MACHINE LEARNING, NUCLEAR PHYSICS, AND ALGORITHM DEVELOPMENT FOR DATA ANALYSIS IN NUCLEAR RESEARCH

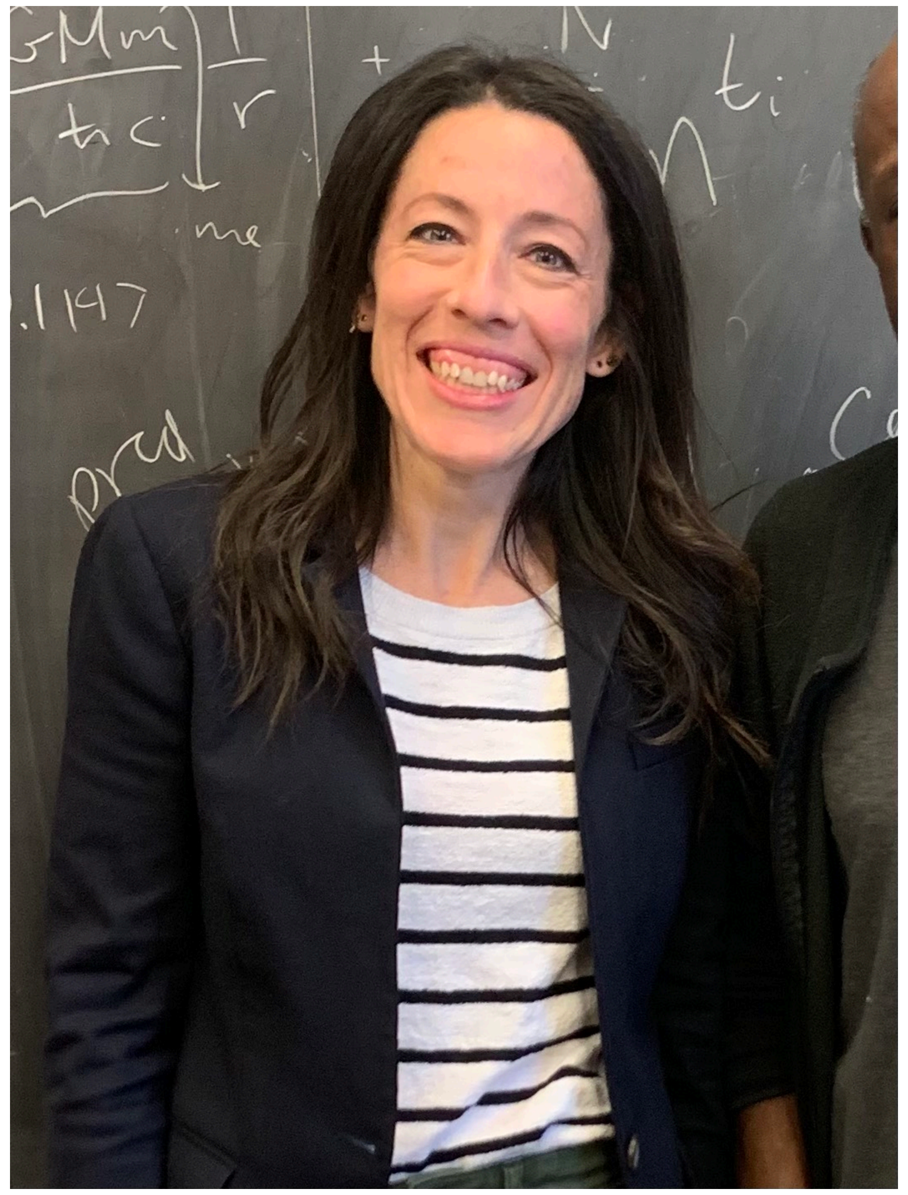
MICHELLE KUCHERA
DAVIDSON COLLEGE

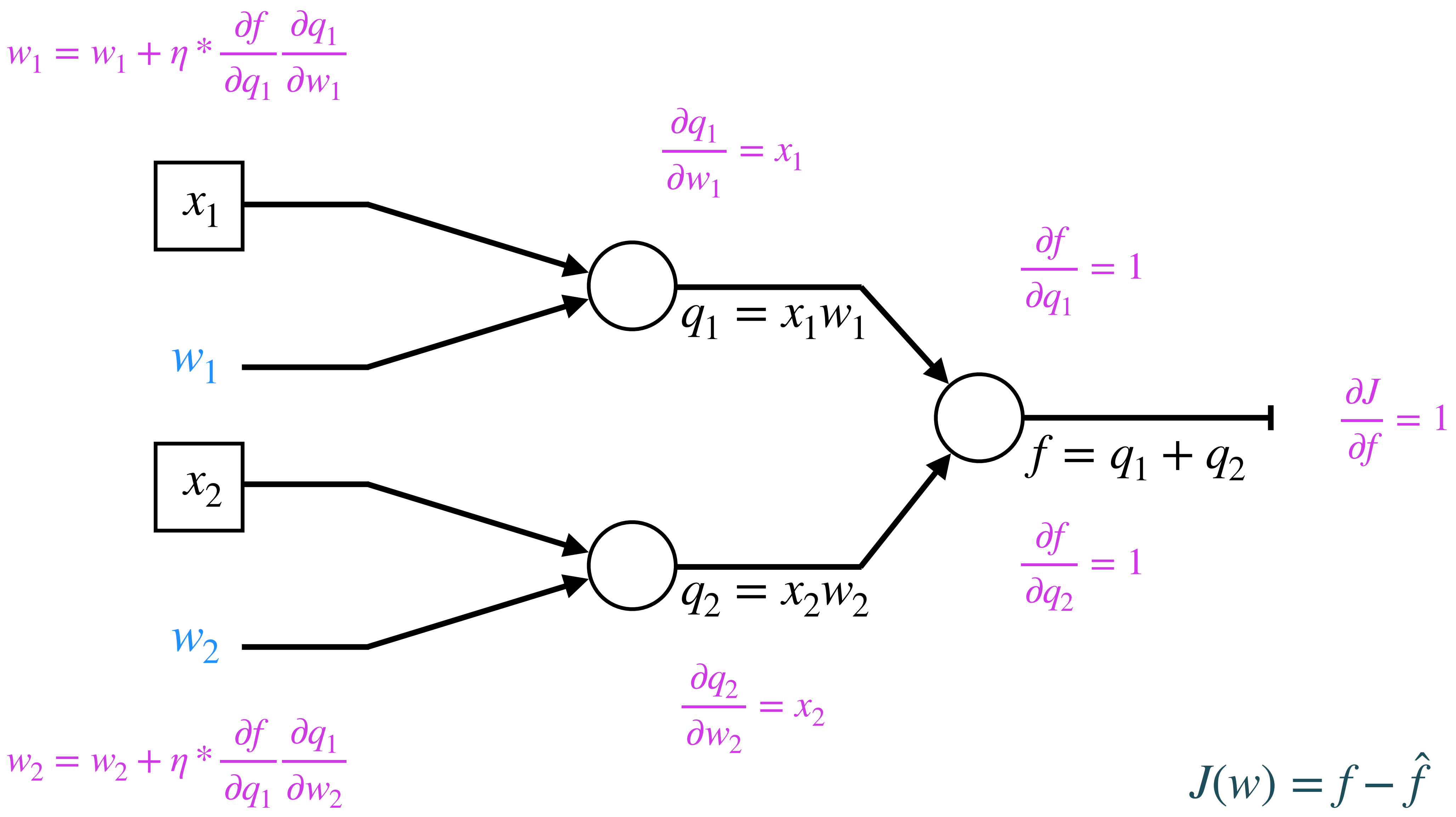
JOINT ICTP-IAEA WORKSHOP ADVANCED SCHOOL ON COMPUTATIONAL NUCLEAR SCIENCE

23 MAY 2022

MICHELLE KUCHERA

B.S., M.S. PHYSICS
M.S., PH.D. COMPUTATIONAL SCIENCE





ALPhA DAVIDSON COLLEGE

CERN

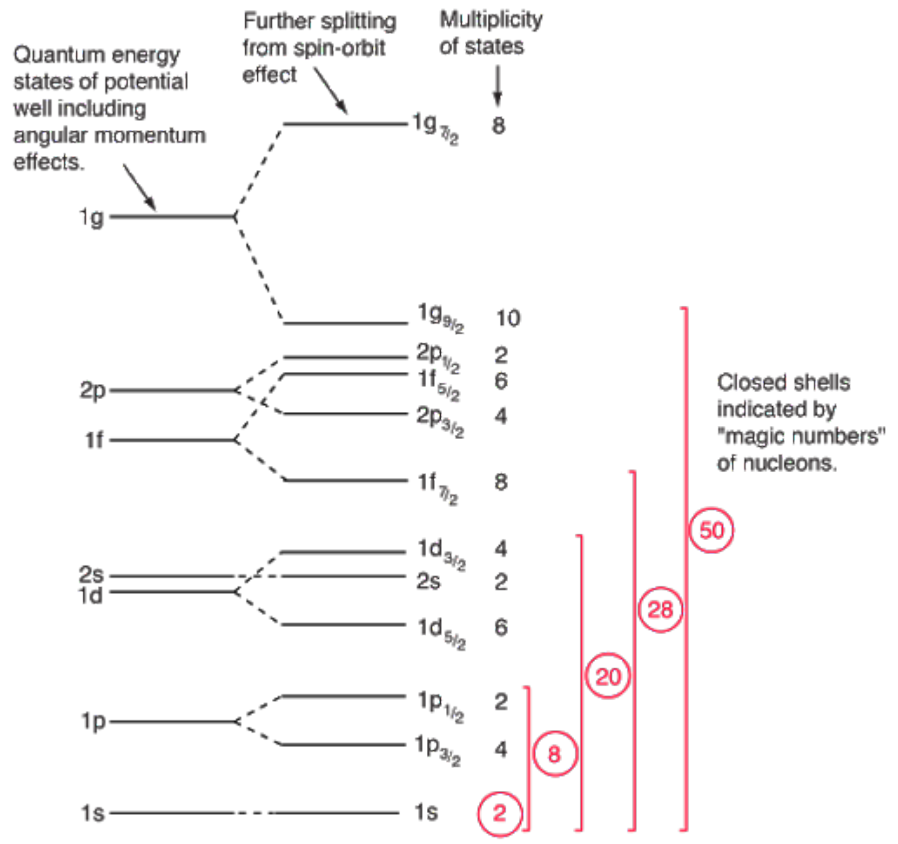
CMS Compact Muon Solenoid

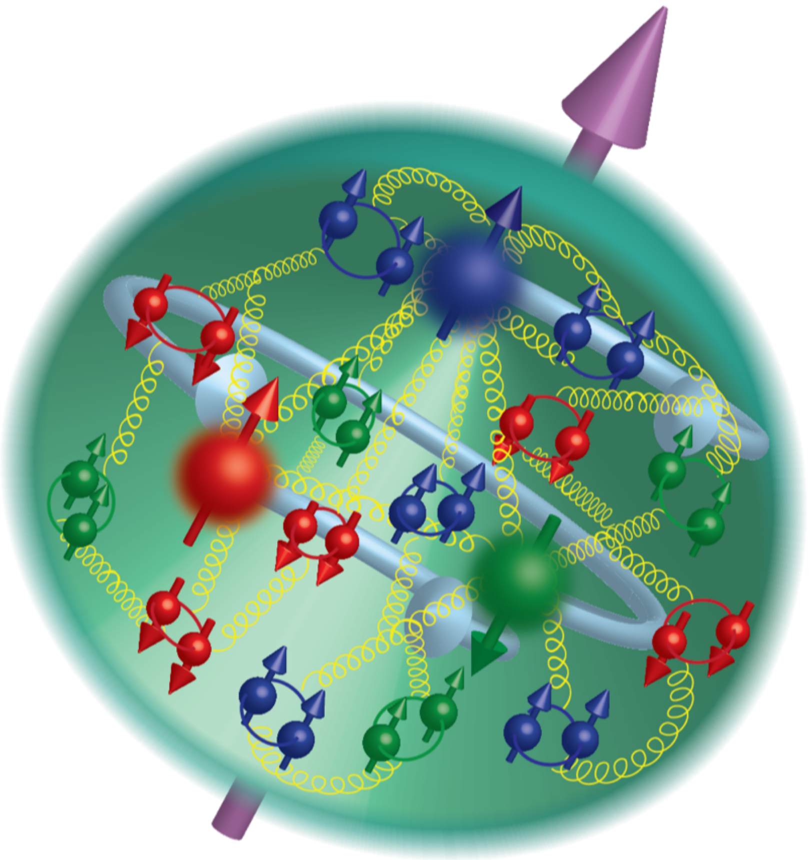
Jefferson Lab

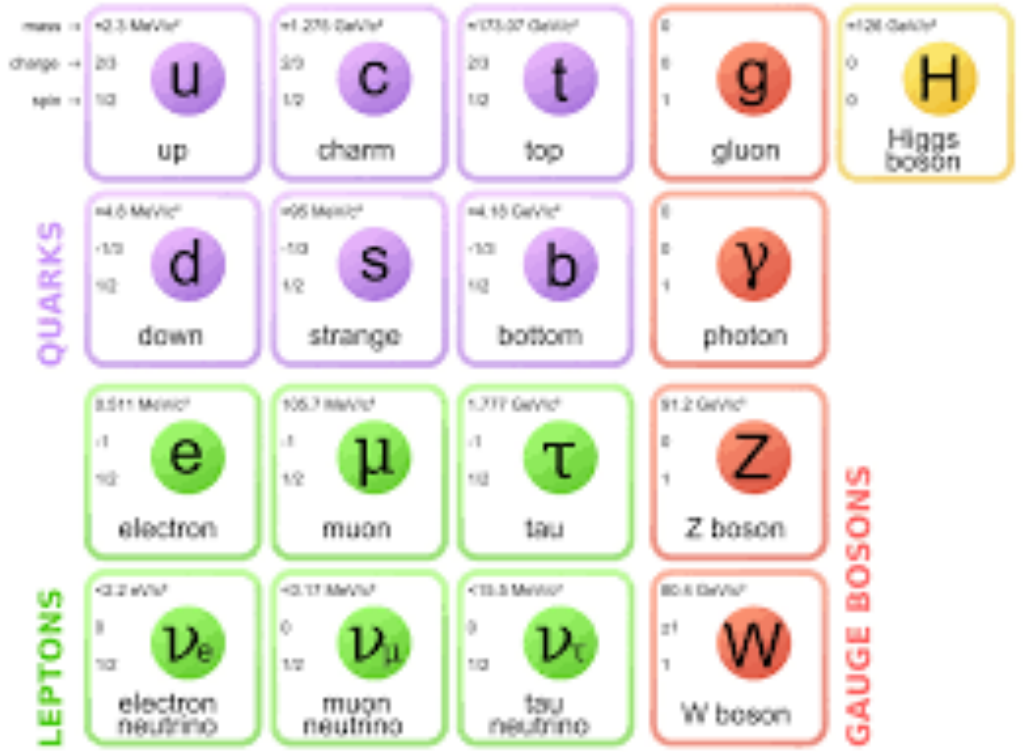
FRIB

DAVIDSON

NSCL





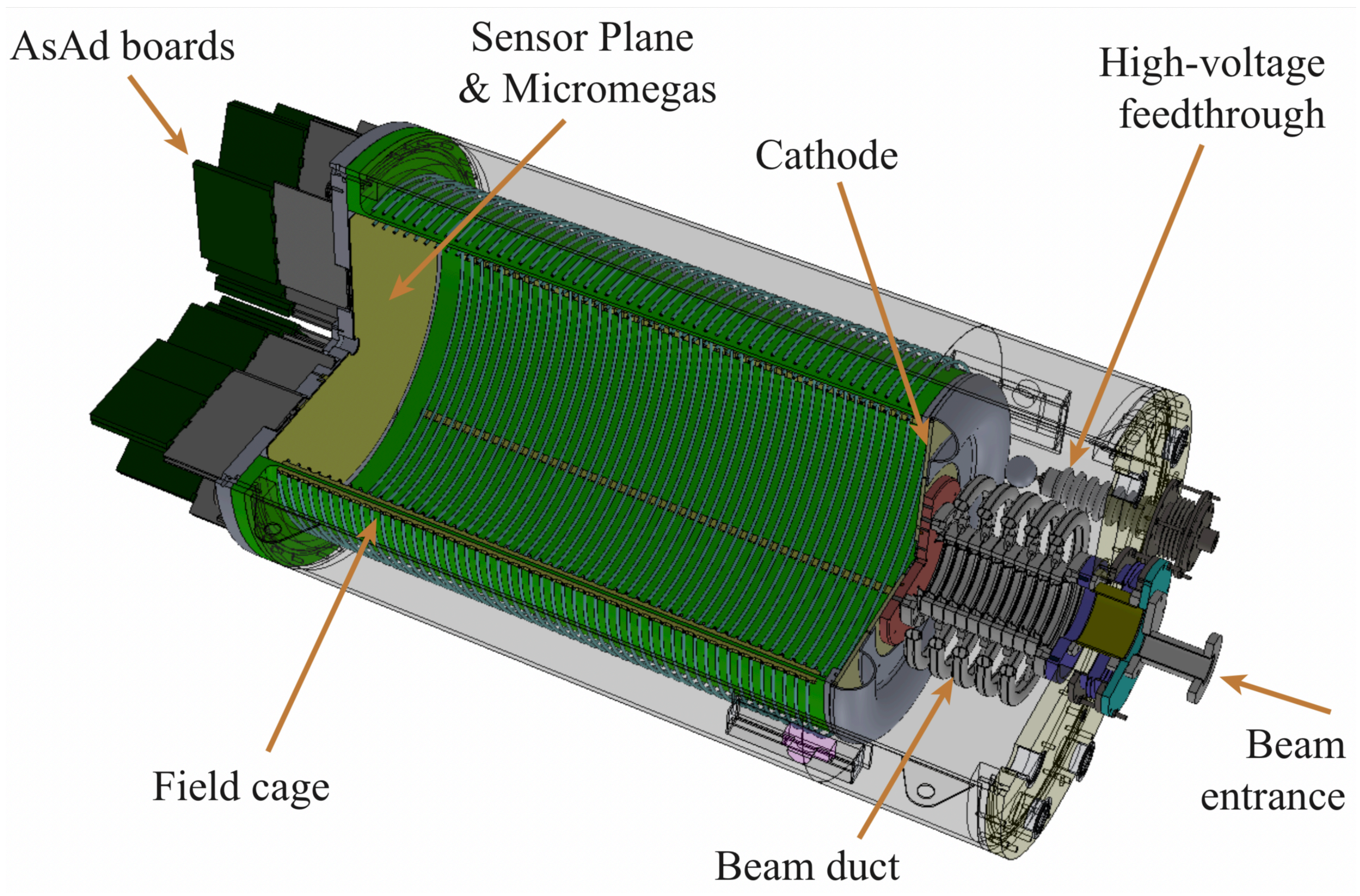


FRIB

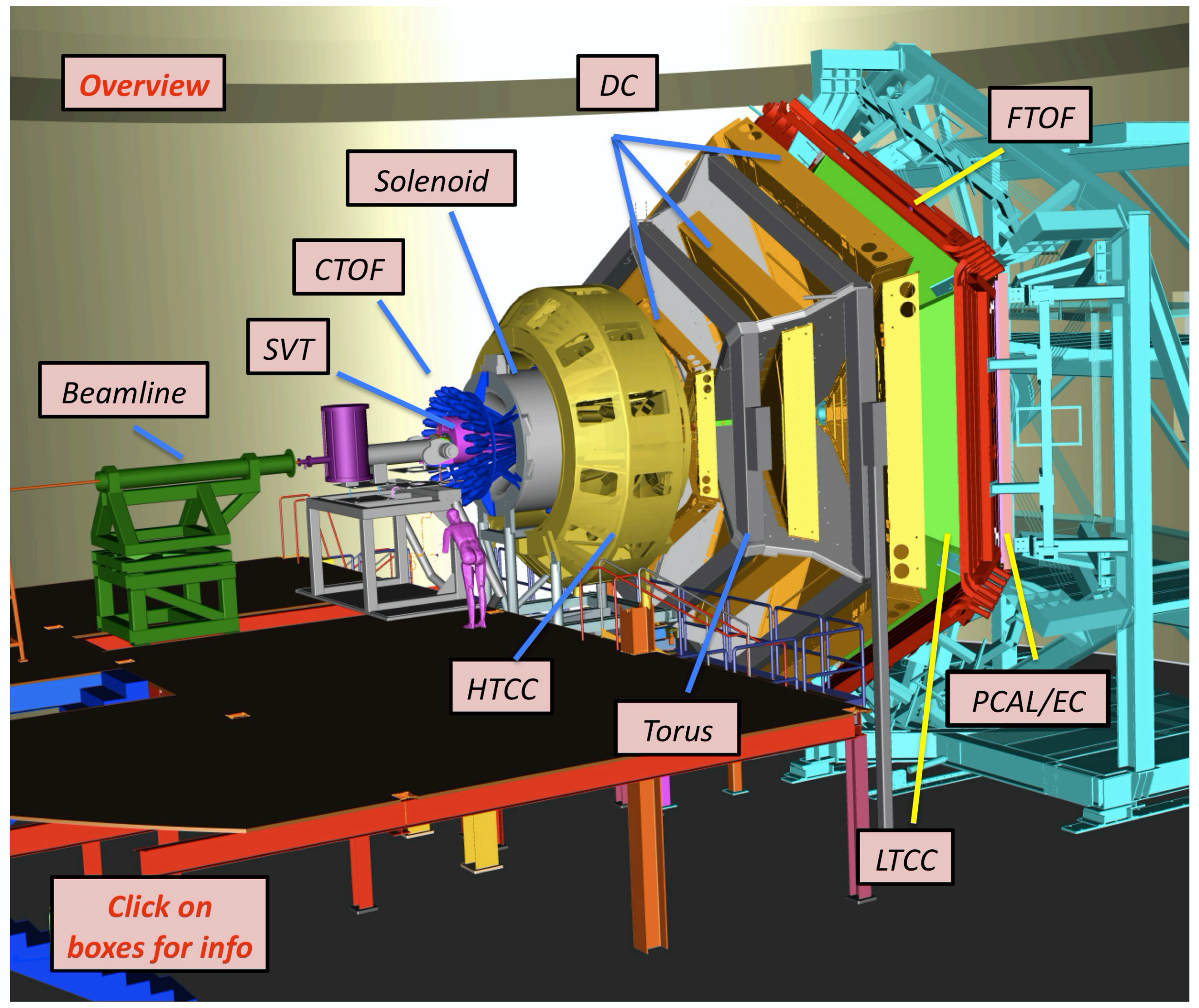
Jefferson Lab

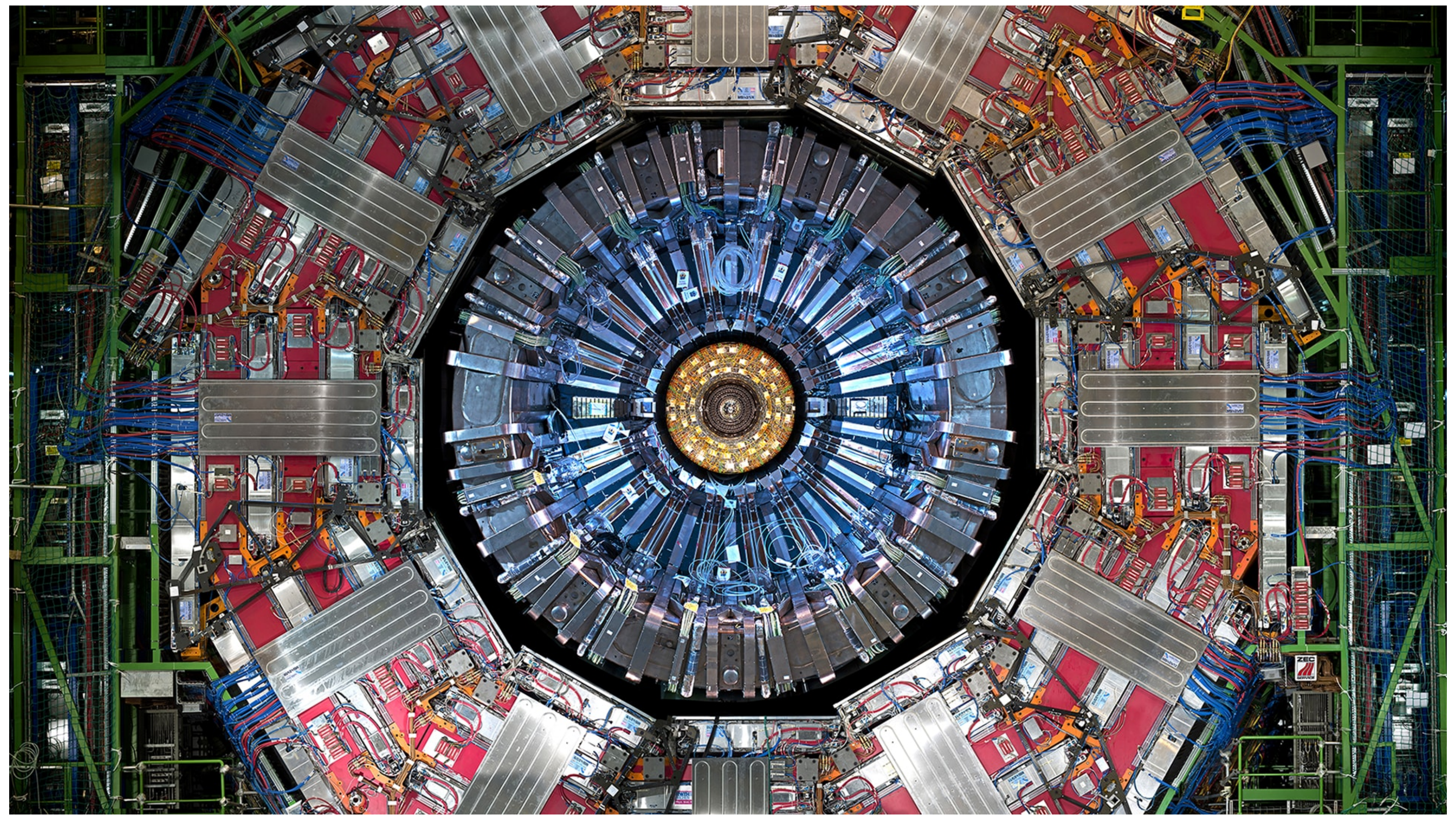
CERN

EXPERIMENTAL DATA



J. BRADT ET. AL., *NUCLEAR INSTRUMENTS AND METHODS*, 2017.





FRIB

AT-TPC

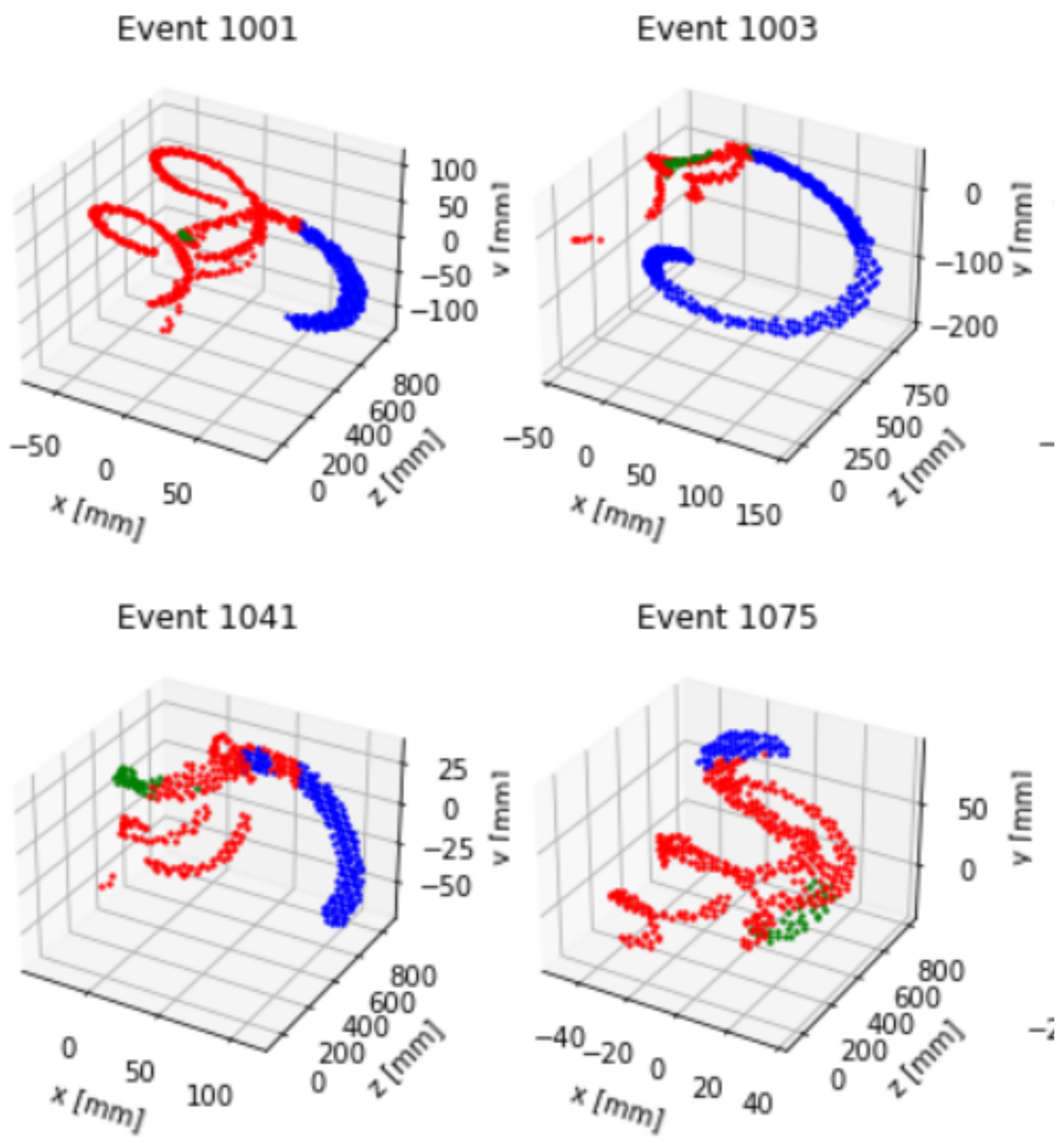
Jefferson Lab

CLAS 12

CERN

CMS

EXPERIMENTAL DATA

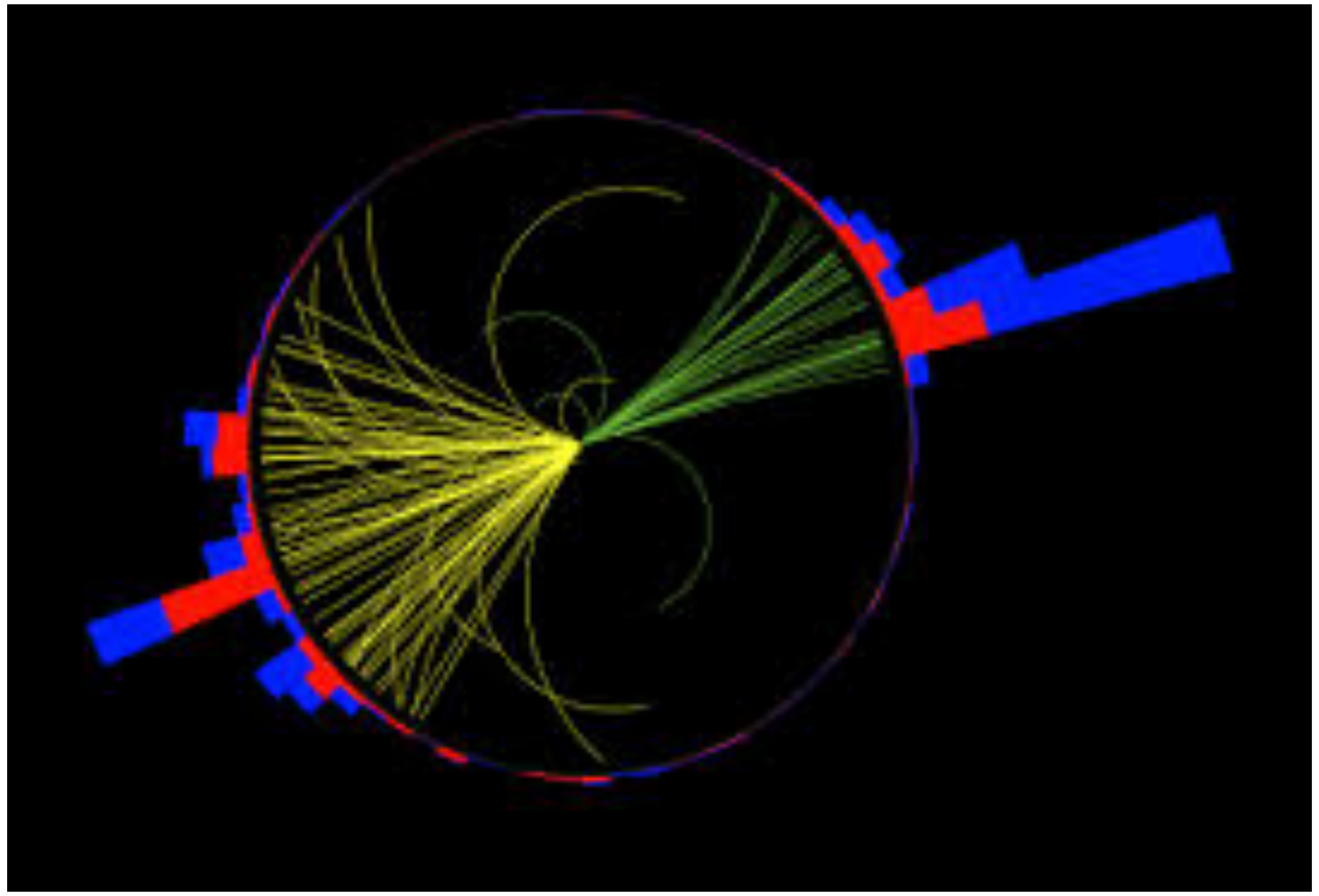


FRIB

AT-TPC

Jefferson Lab

CLAS 12

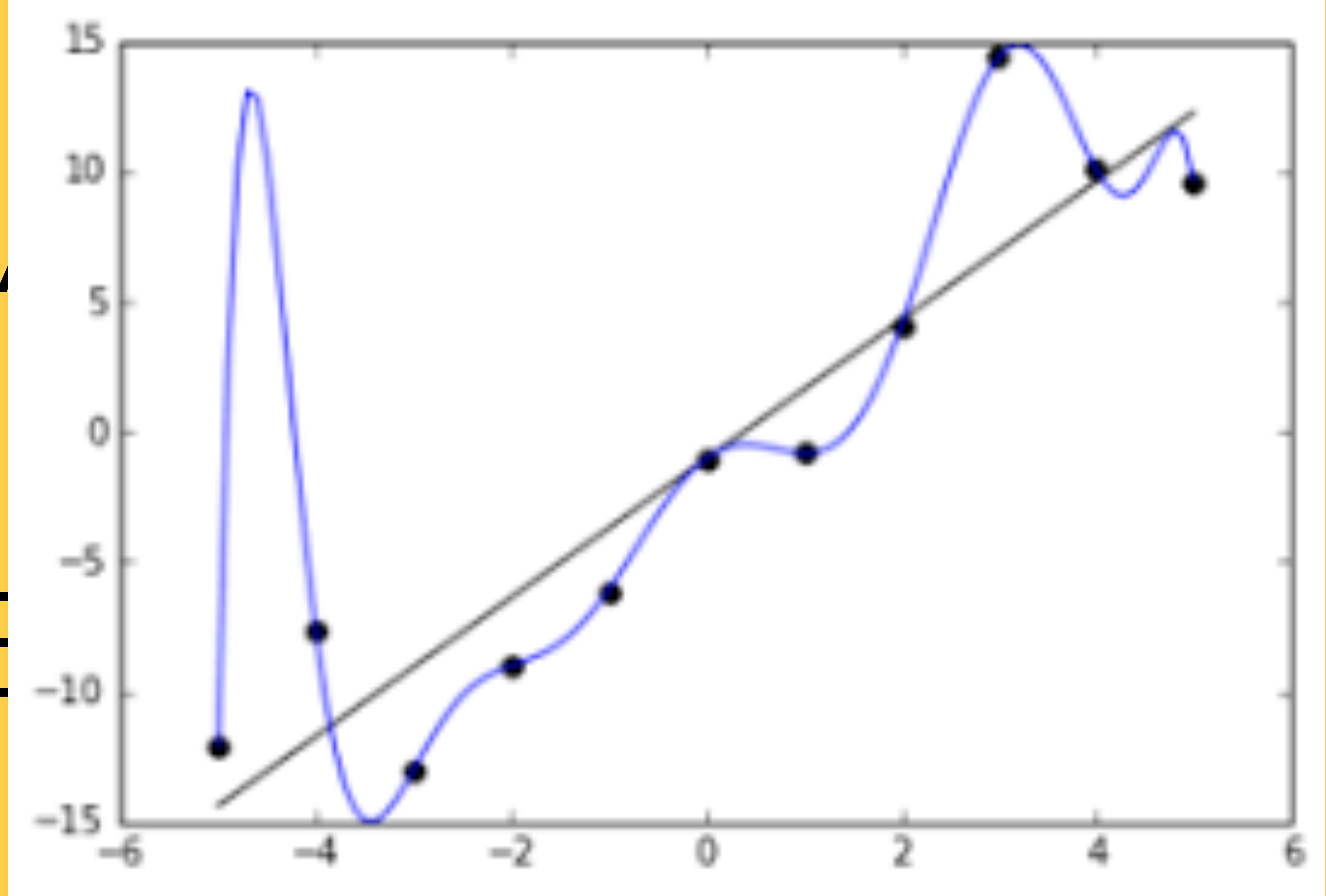


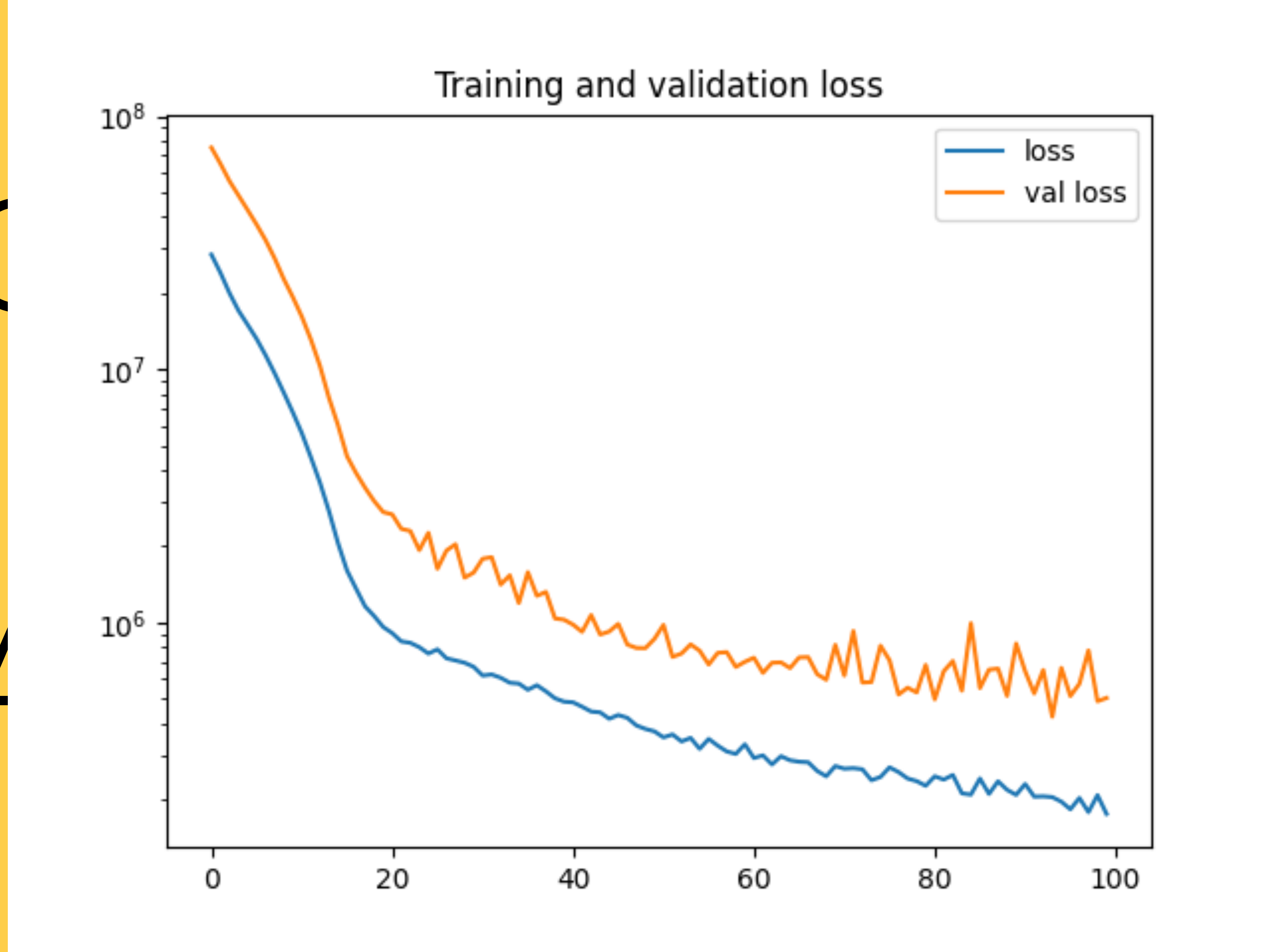
CERN

CMS

M A … I N G

L E … D A


15
10
5
0
−5
−10
−15
−6
−4
−2
0
2
4
6


Training and validation loss
loss
val loss
10^8
10^7
10^6
0
20
40
60
80
100

NEURON

MATHEMATICS

ELSEVIER

Neural Networks
Volume 4, Issue 2, 1991, Pages 251-257

Approximation capabilities of multilayer feedforward networks

Kurt Hornik

Show more

Share Cite

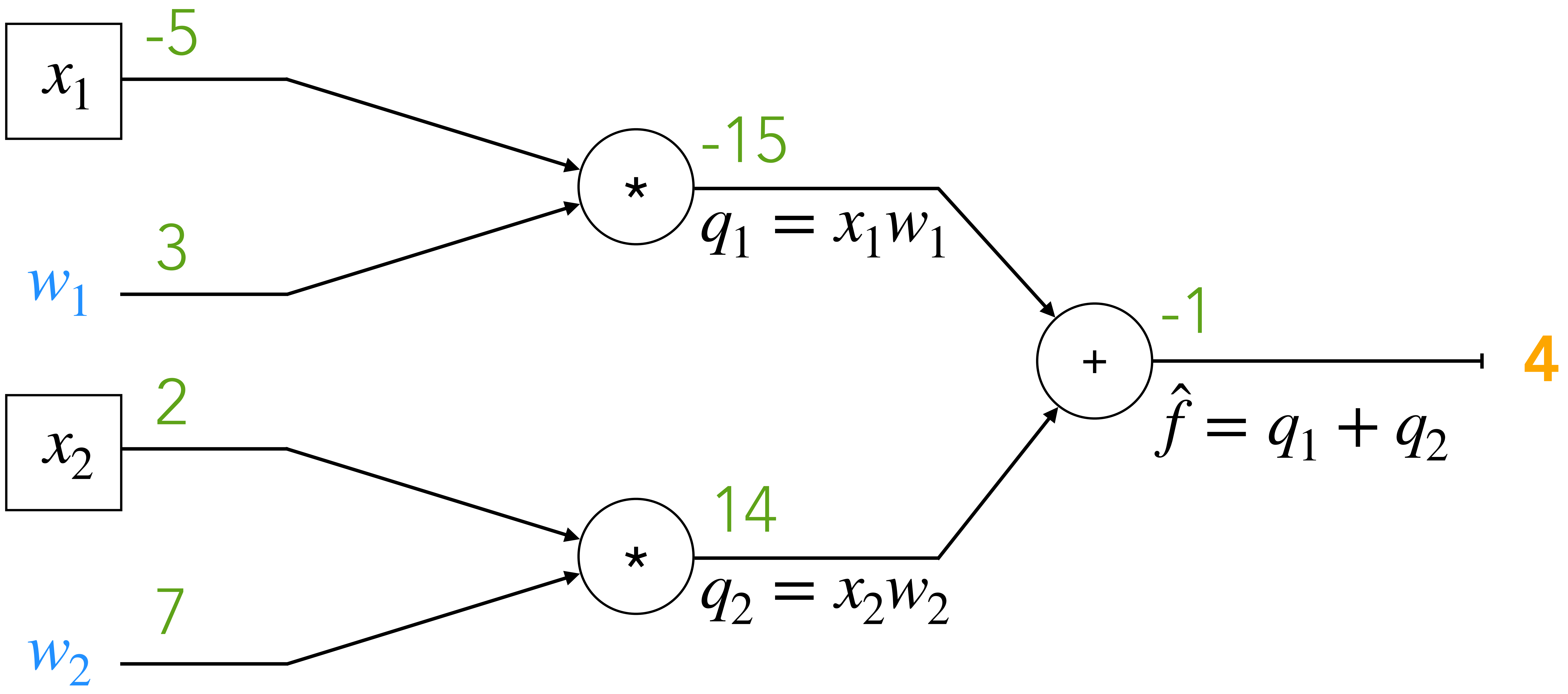
https://doi.org/10.1016/0893-6080(91)90009-T

Abstract

We show that standard multilayer feedforward networks with as few as a single hidden layer and arbitrary bounded and nonconstant activation function are universal approximators with respect to $L^p(\mu)$ performance criteria, for arbitrary finite input environment measures μ, provided only that sufficiently many hidden units are available. If the activation function is continuous, bounded and nonconstant, then continuous mappings can be learned uniformly over compact input sets. We also give very general conditions ensuring that networks with sufficiently smooth activation functions are capable of arbitrarily accurate approximation to a function and its derivatives.

MATHEMATICS

COMPUTATIONAL GRAPH

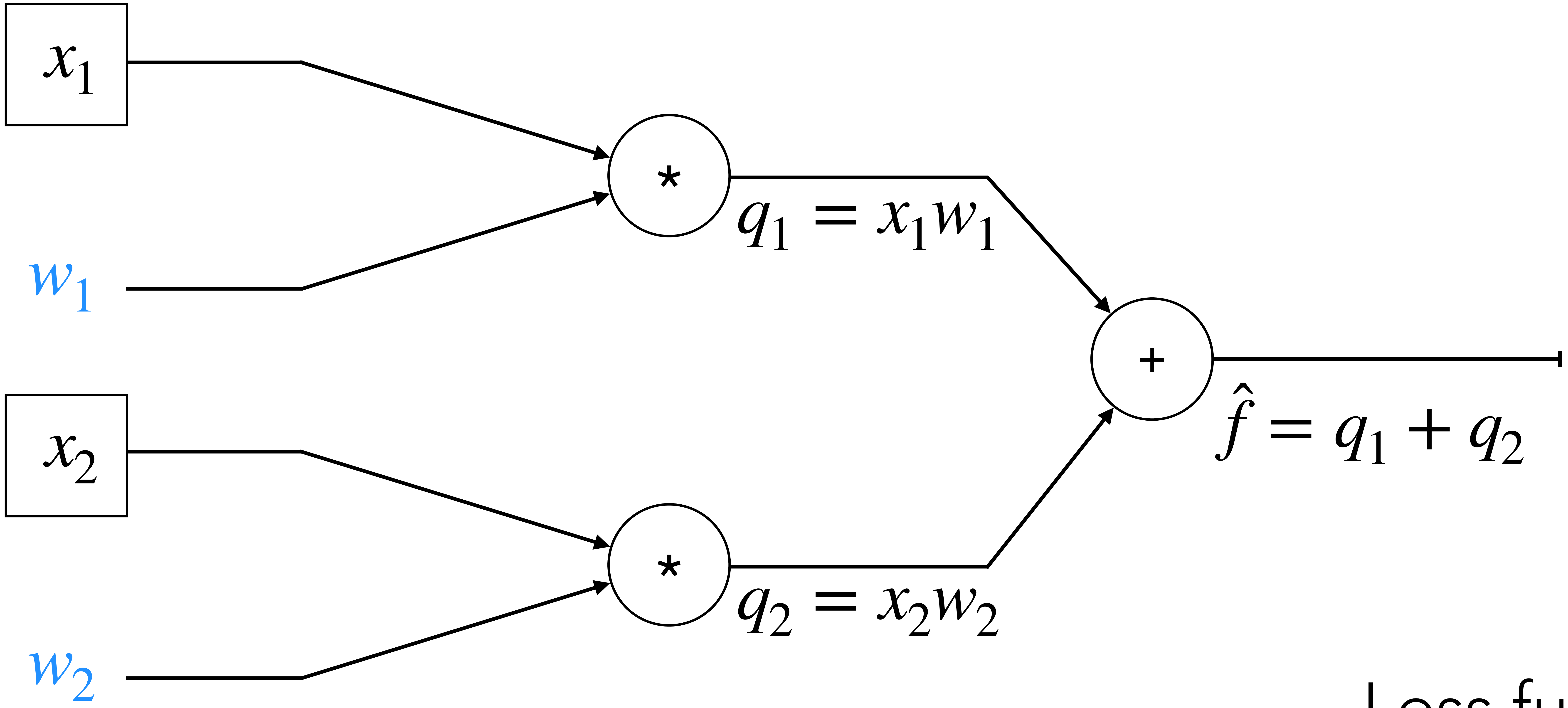


$$\hat{f} = x_1 w_1 + x_2 w_2$$

MACHINE LEARNING

SUPERVISED LEARNING

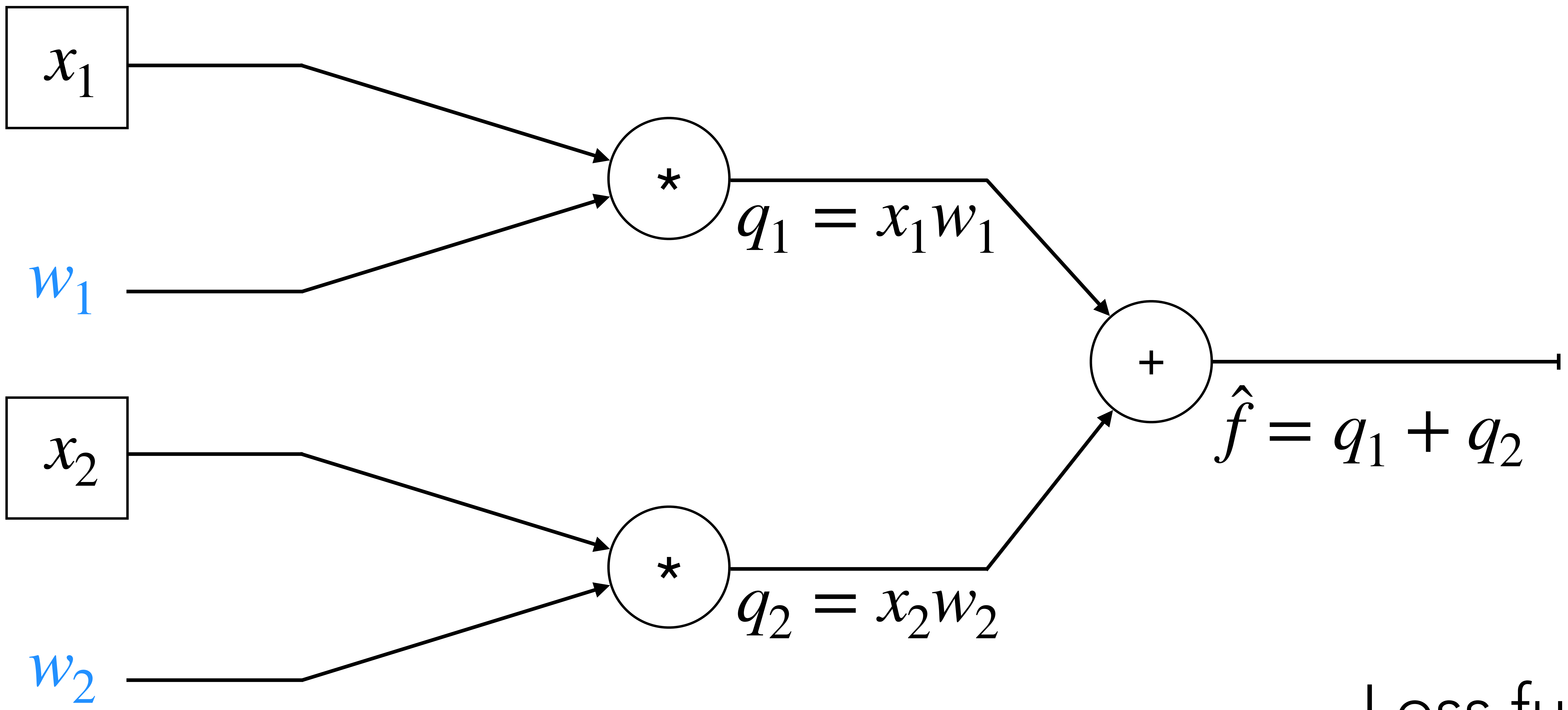
REGRESSION



Loss function

$$\hat{f} = x_1 w_1 + x_2 w_2 \qquad J(w) = \hat{f} - f$$

SUPERVISED LEARNING



Loss function

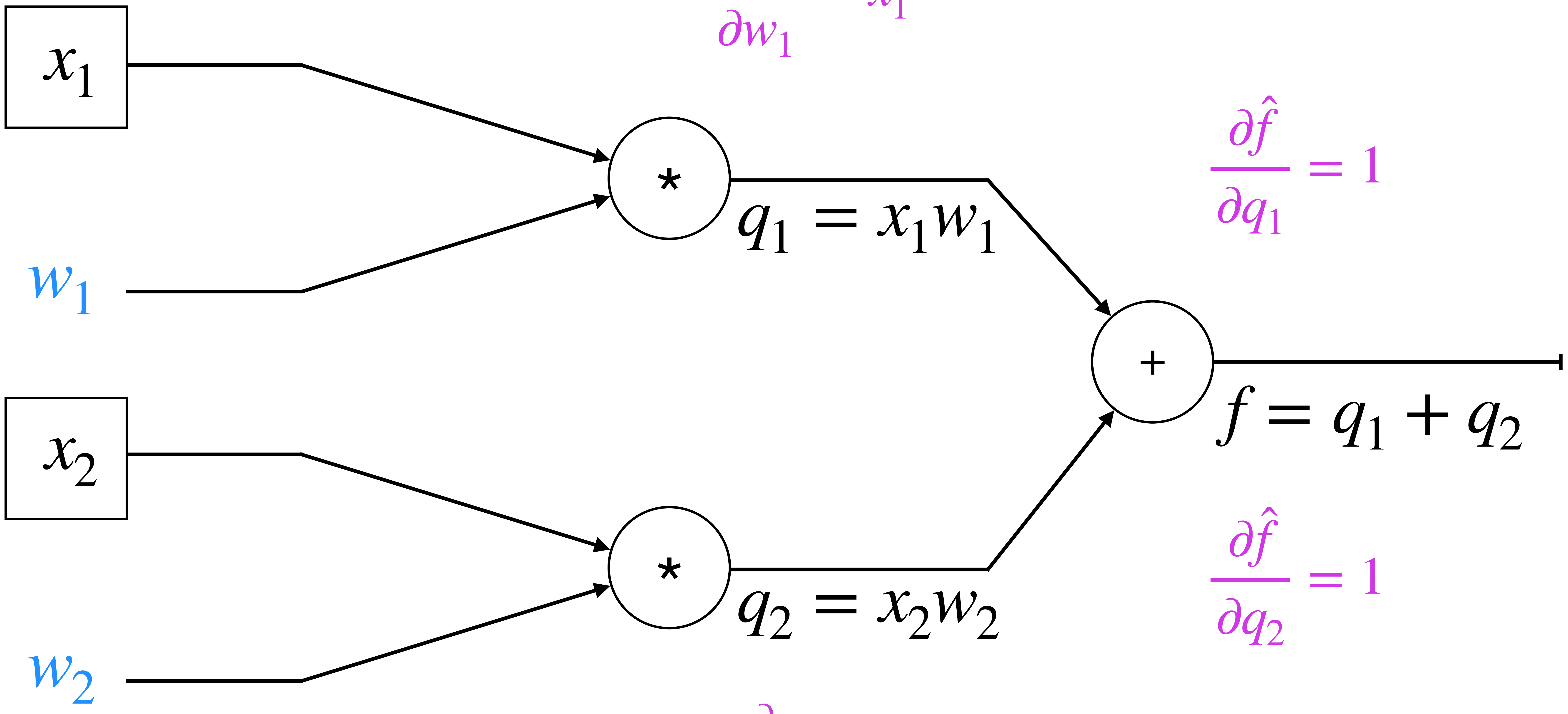
$\hat{f} = x_1 w_1 + x_2 w_2$

$J(w) = \hat{f} - f$

BACKPROPAGATION

$$w_1 = w_1 + \eta * \frac{\partial J}{\partial \hat{f}} \frac{\partial \hat{f}}{\partial q_1} \frac{\partial q_1}{\partial w_1}$$

$$\frac{\partial q_1}{\partial w_1} = x_1$$



$$\frac{\partial \hat{f}}{\partial q_1} = 1$$

$$\frac{\partial J}{\partial \hat{f}} = 1$$

$$\frac{\partial \hat{f}}{\partial q_2} = 1$$

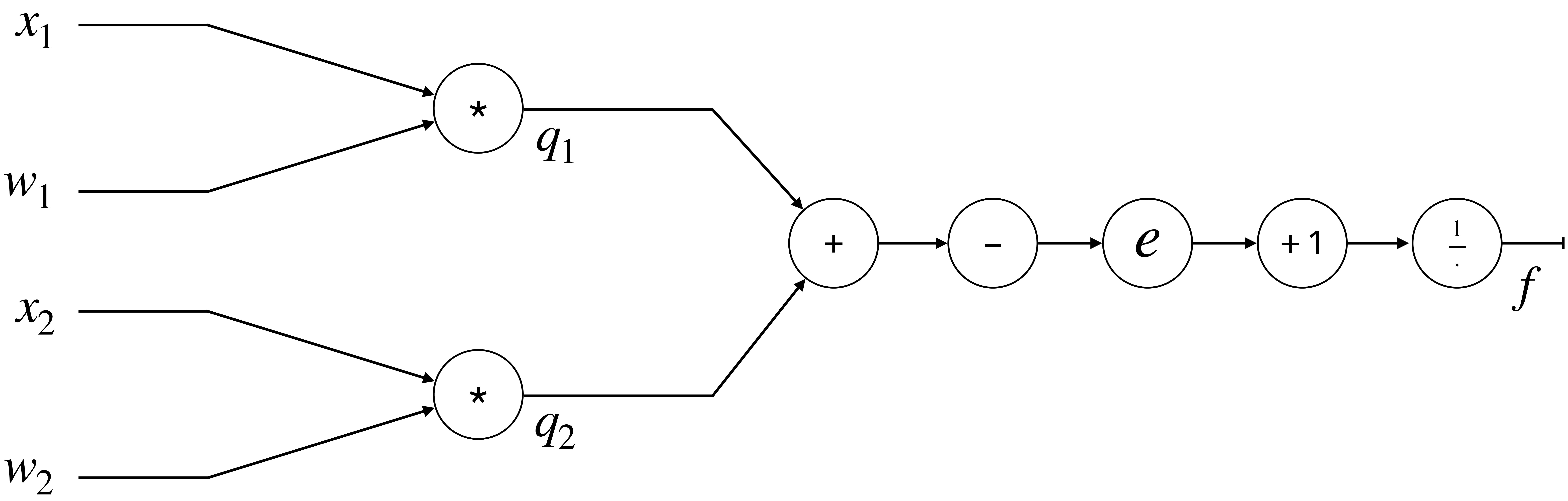
$$\frac{\partial q_2}{\partial w_2} = x_2$$

$$w_2 = w_2 + \eta * \frac{\partial J}{\partial \hat{f}} \frac{\partial \hat{f}}{\partial q_2} \frac{\partial q_2}{\partial w_2}$$

Loss function

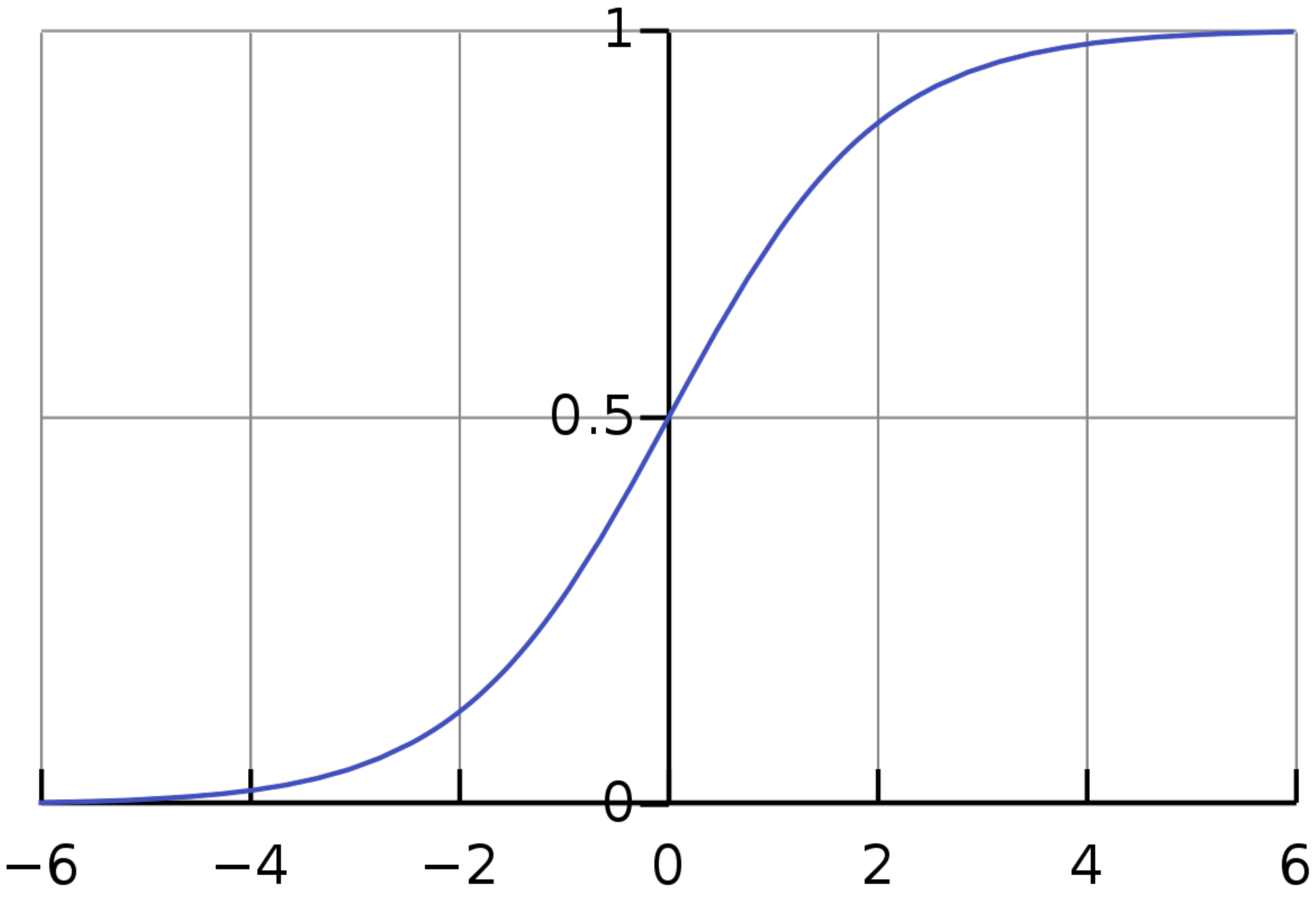
$$J(w) = \hat{f} - f$$

LOGISTIC REGRESSION

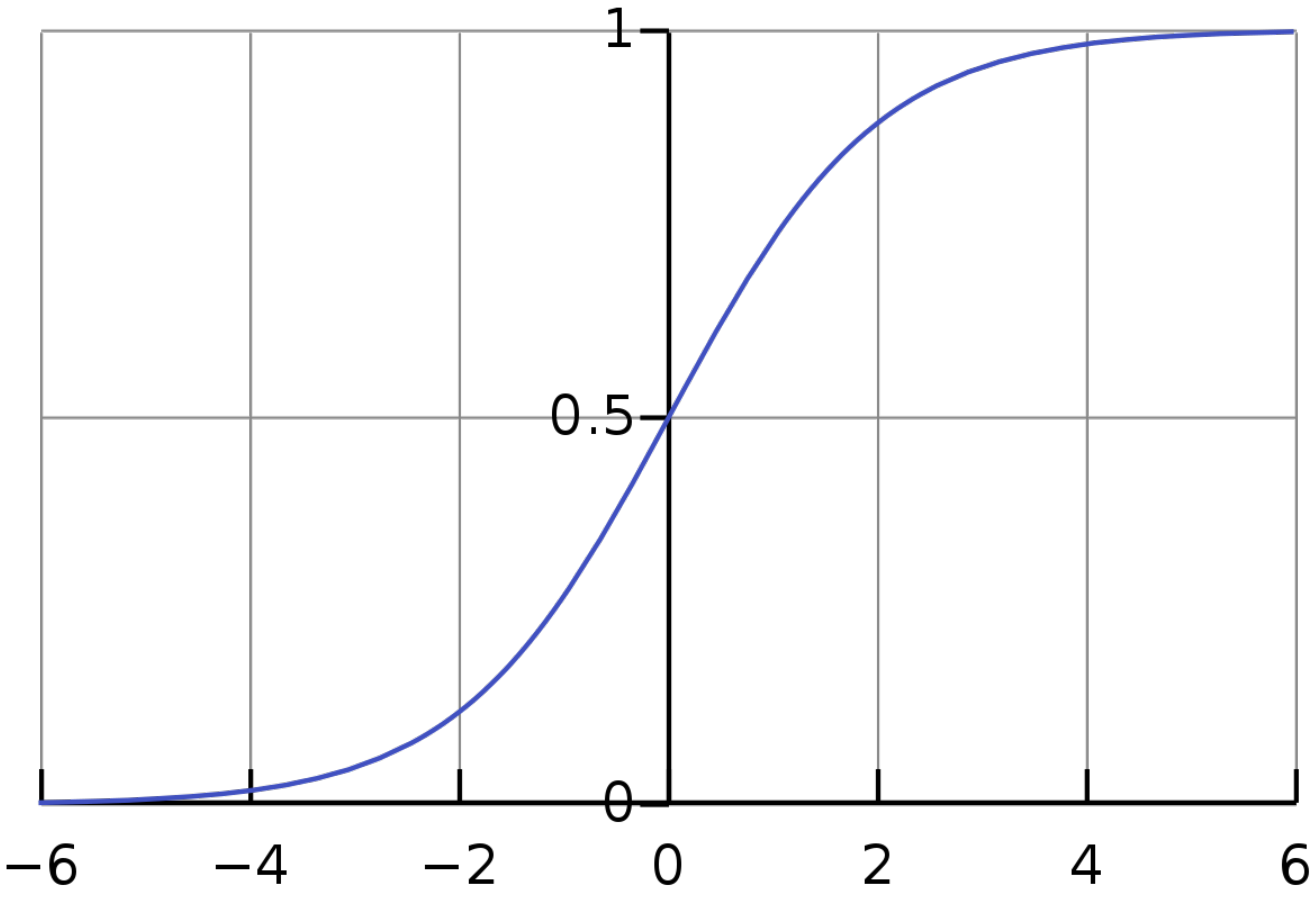


$$f = \frac{1}{1 + e^{-(x_1 w_1 + x_2 w_2)}}$$

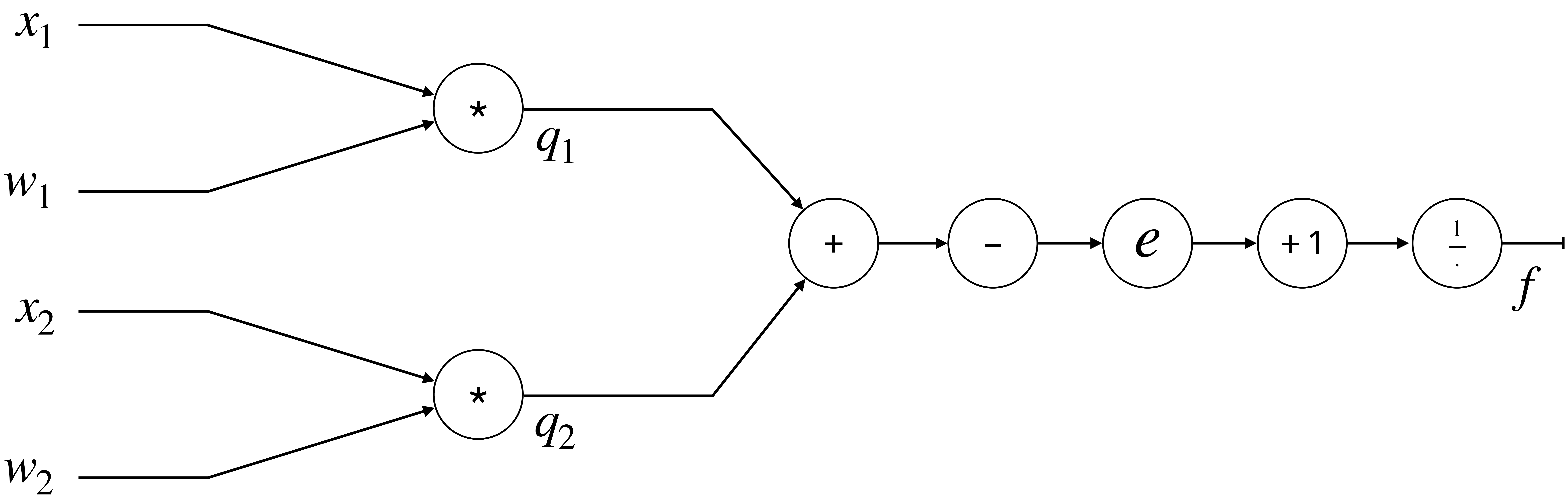
LOGISTIC REGRESSION



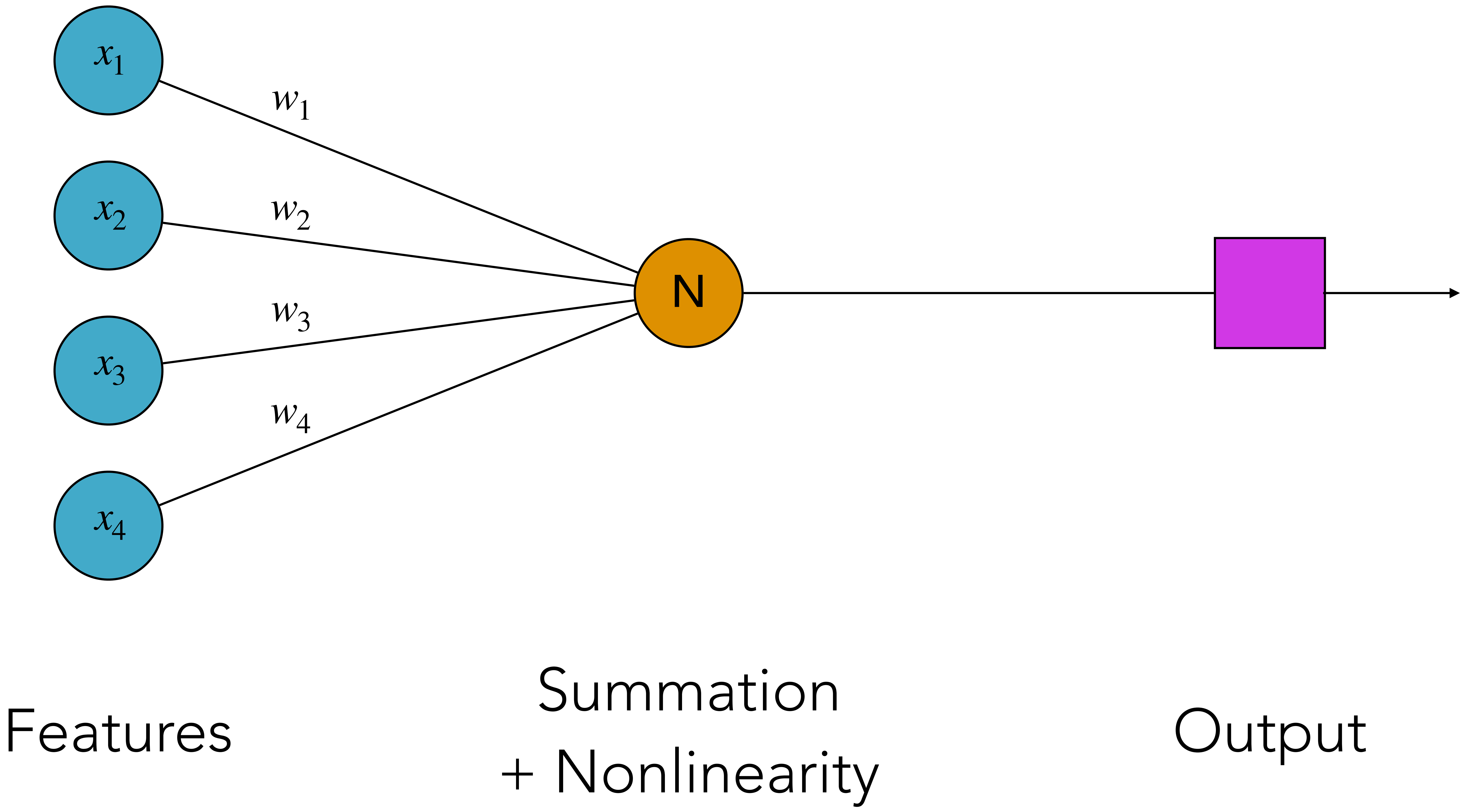
BINARY CLASSIFICATION

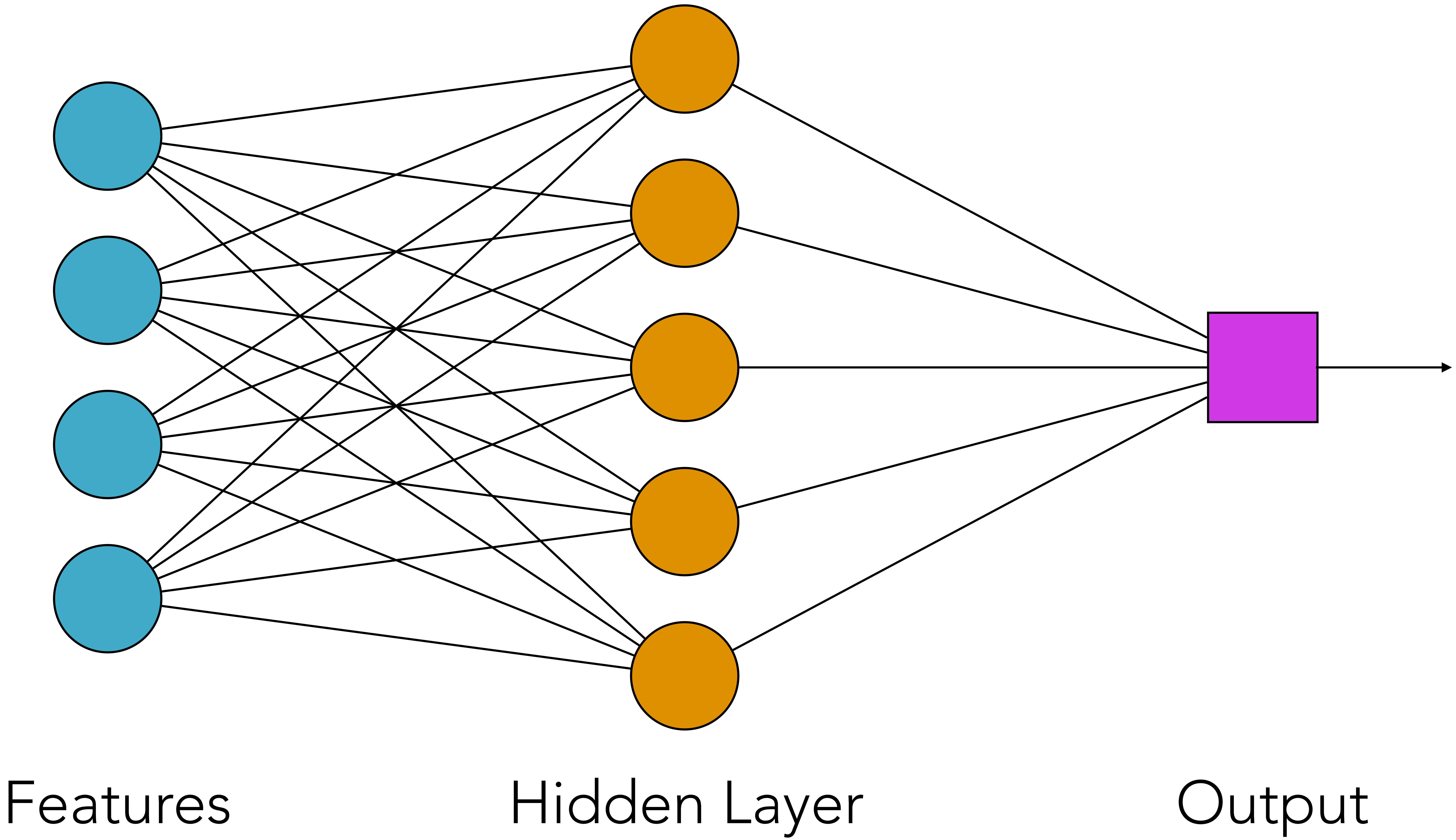


LOGISTIC REGRESSION



$$f = \frac{1}{1 + e^{-(x_1 w_1 + x_2 w_2)}}$$

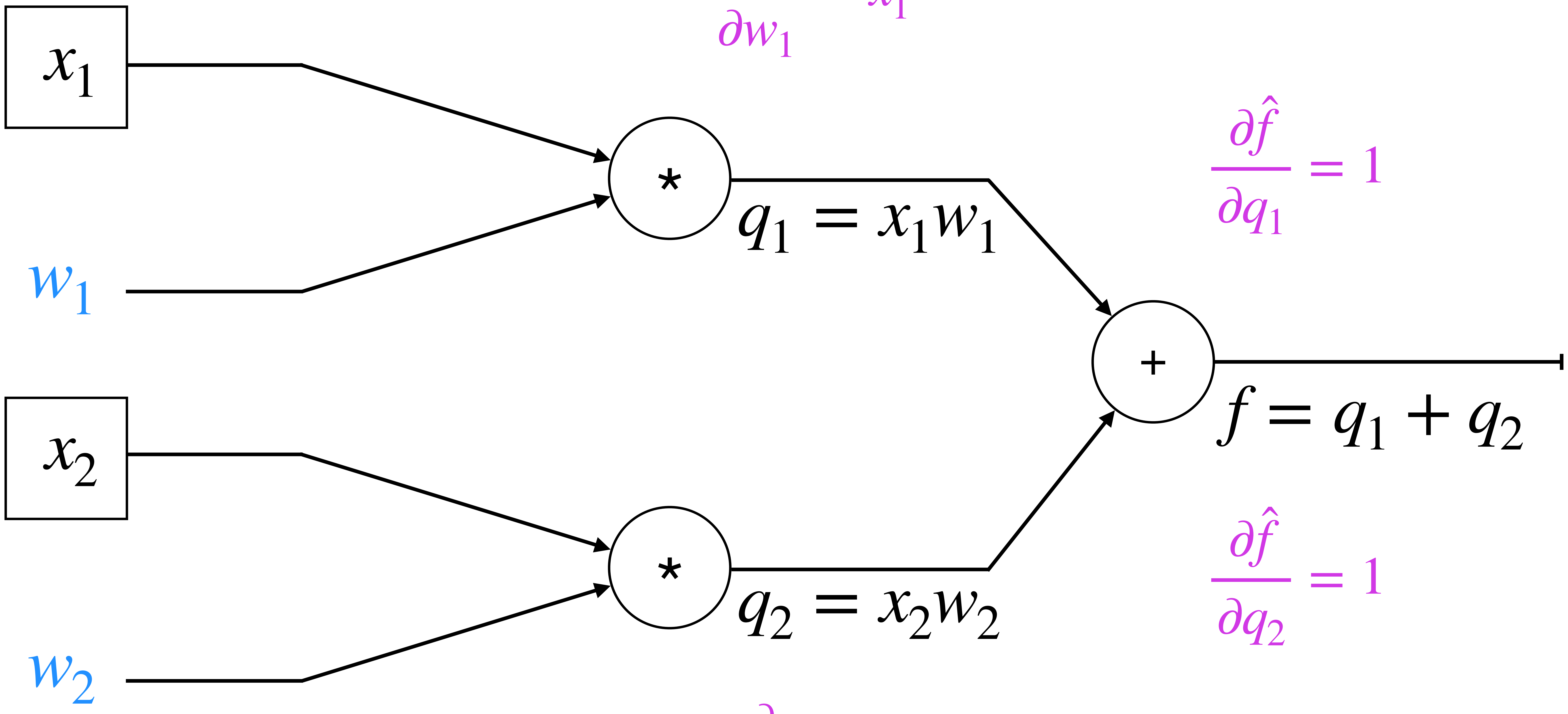

x_1
x_2
x_3
x_4
w_1
w_2
w_3
w_4
N
Features
Summation
+ Nonlinearity
Output


Features
Hidden Layer
Output

BACKPROPAGATION

$$w_1 = w_1 + \eta * \frac{\partial J}{\partial \hat{f}} \frac{\partial \hat{f}}{\partial q_1} \frac{\partial q_1}{\partial w_1}$$

$$\frac{\partial q_1}{\partial w_1} = x_1$$



$$\frac{\partial \hat{f}}{\partial q_1} = 1$$

$$\frac{\partial J}{\partial \hat{f}} = 1$$

$$\frac{\partial \hat{f}}{\partial q_2} = 1$$

$$\frac{\partial q_2}{\partial w_2} = x_2$$

$$w_2 = w_2 + \eta * \frac{\partial J}{\partial \hat{f}} \frac{\partial \hat{f}}{\partial q_2} \frac{\partial q_2}{\partial w_2}$$

Loss function

$$J(w) = \hat{f} - f$$

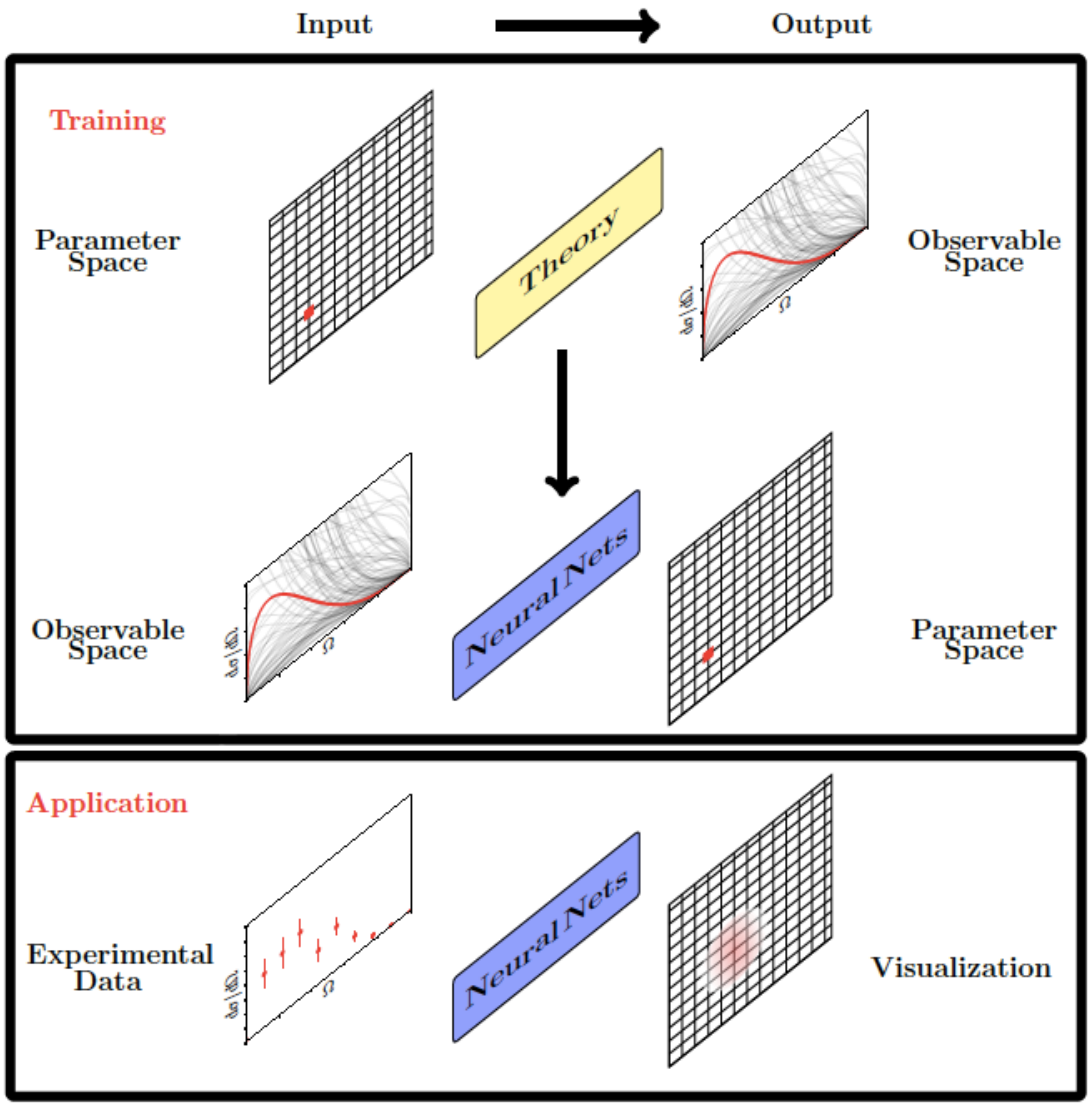
AUTOMATIC DIFFERENTIATION

TensorFlow

PyTorch

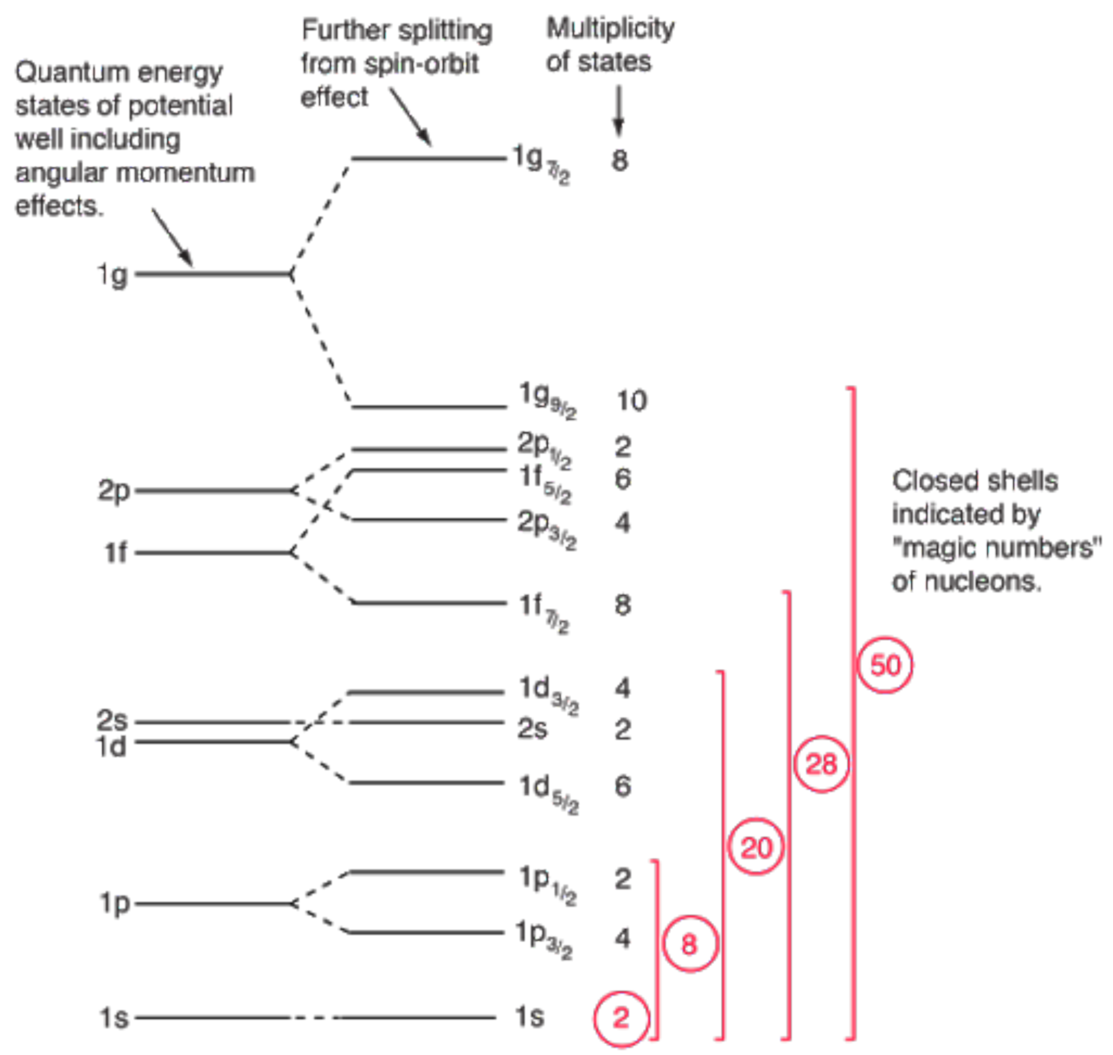
Application 1: How can experimental observables constrain theoretical models?

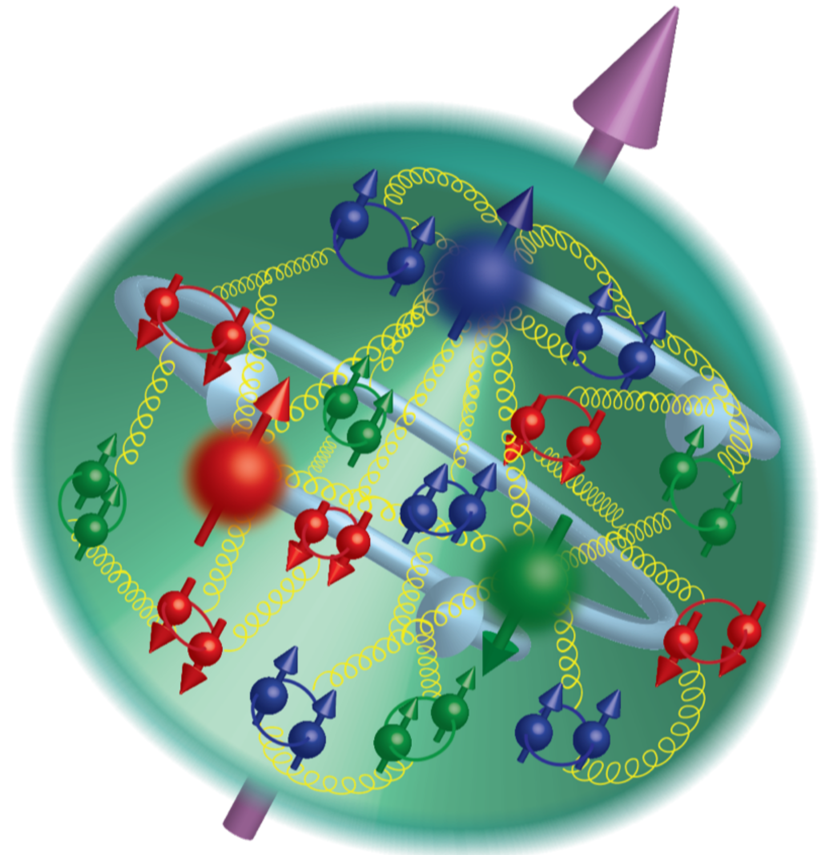
THEORY ⇆ EXPERIMENT

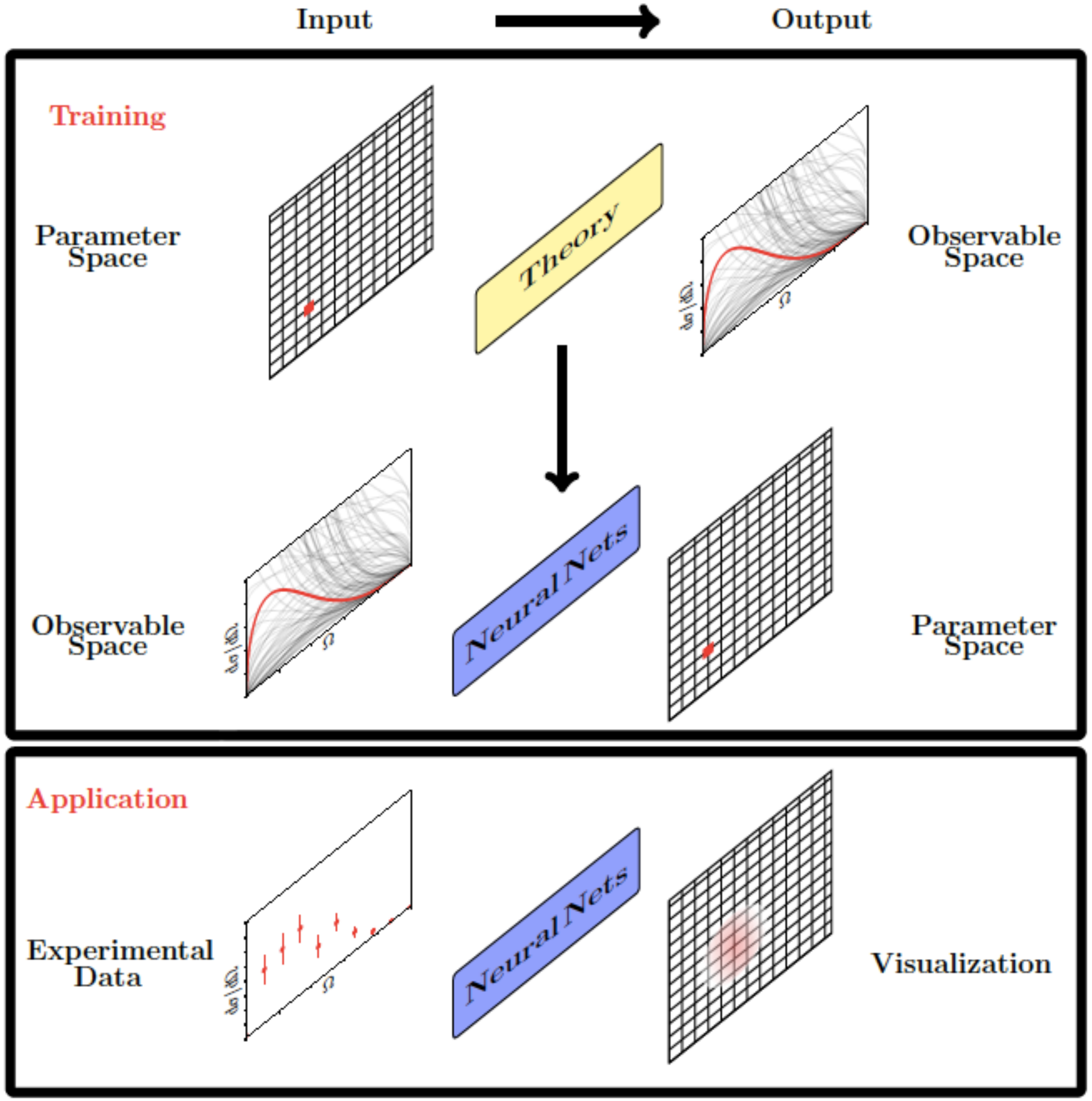


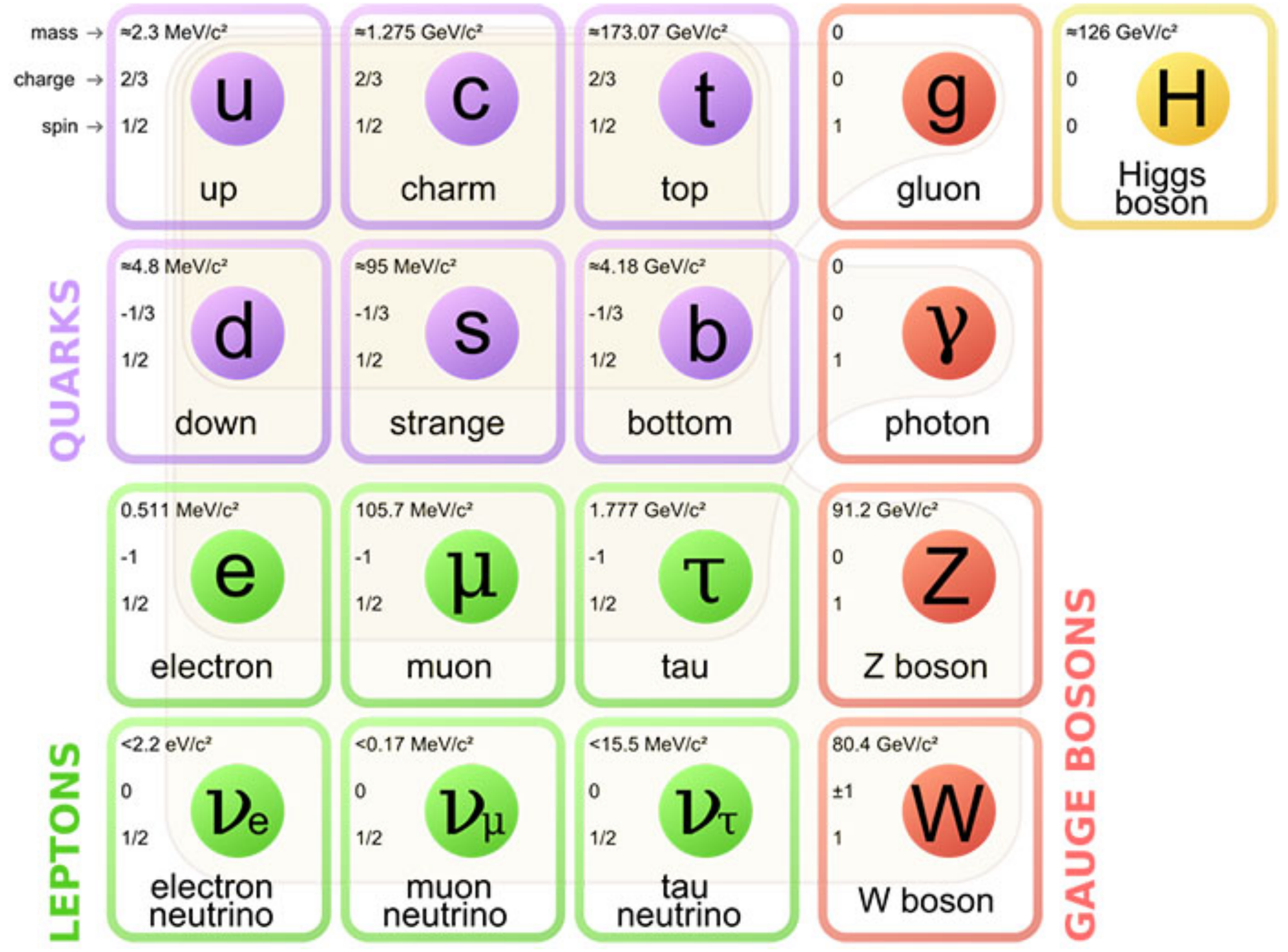
N. SATO, JEFFERSON LAB

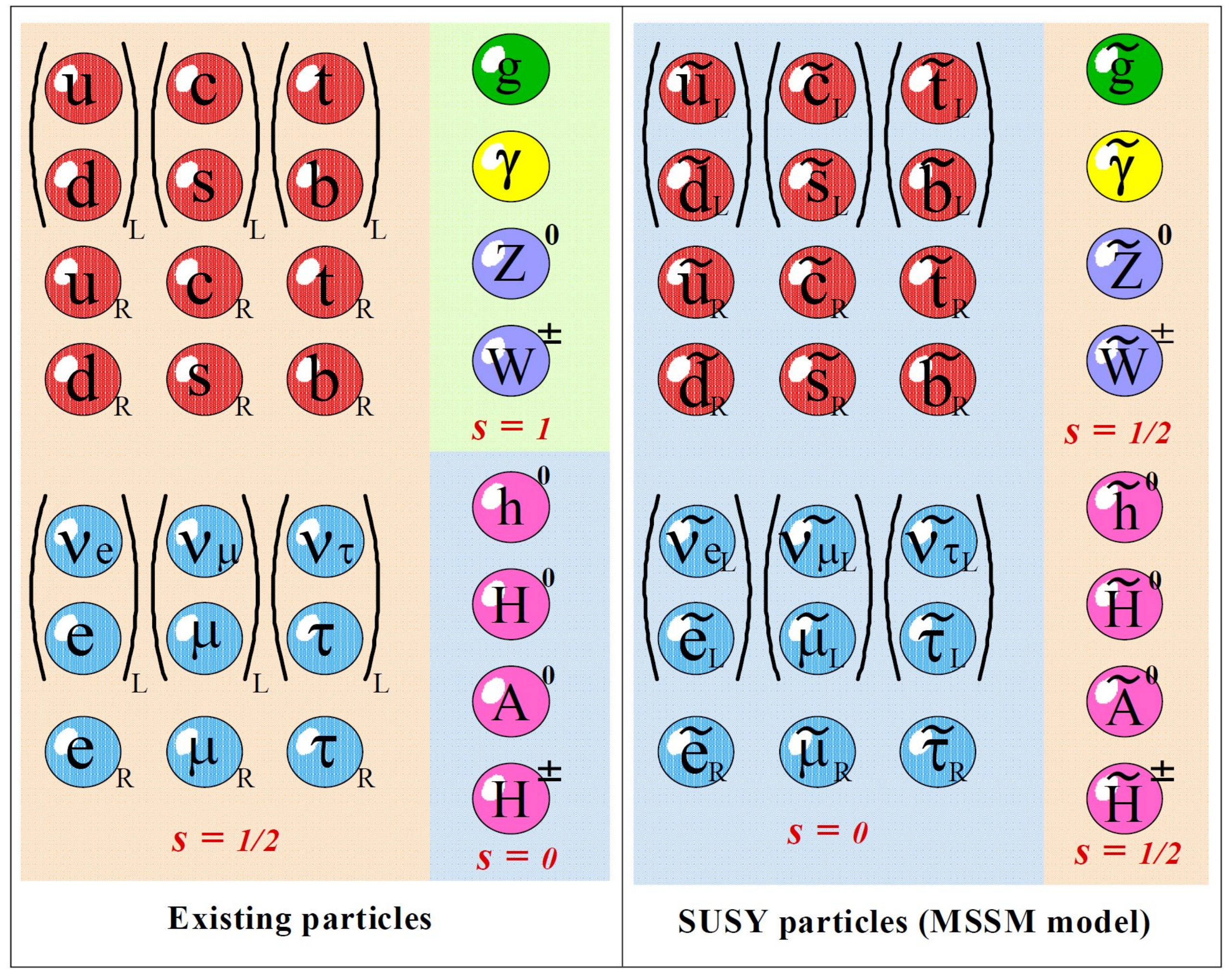
THEORY ⇆ EXPERIMENT











MIXTURE DENSITY NETWORK

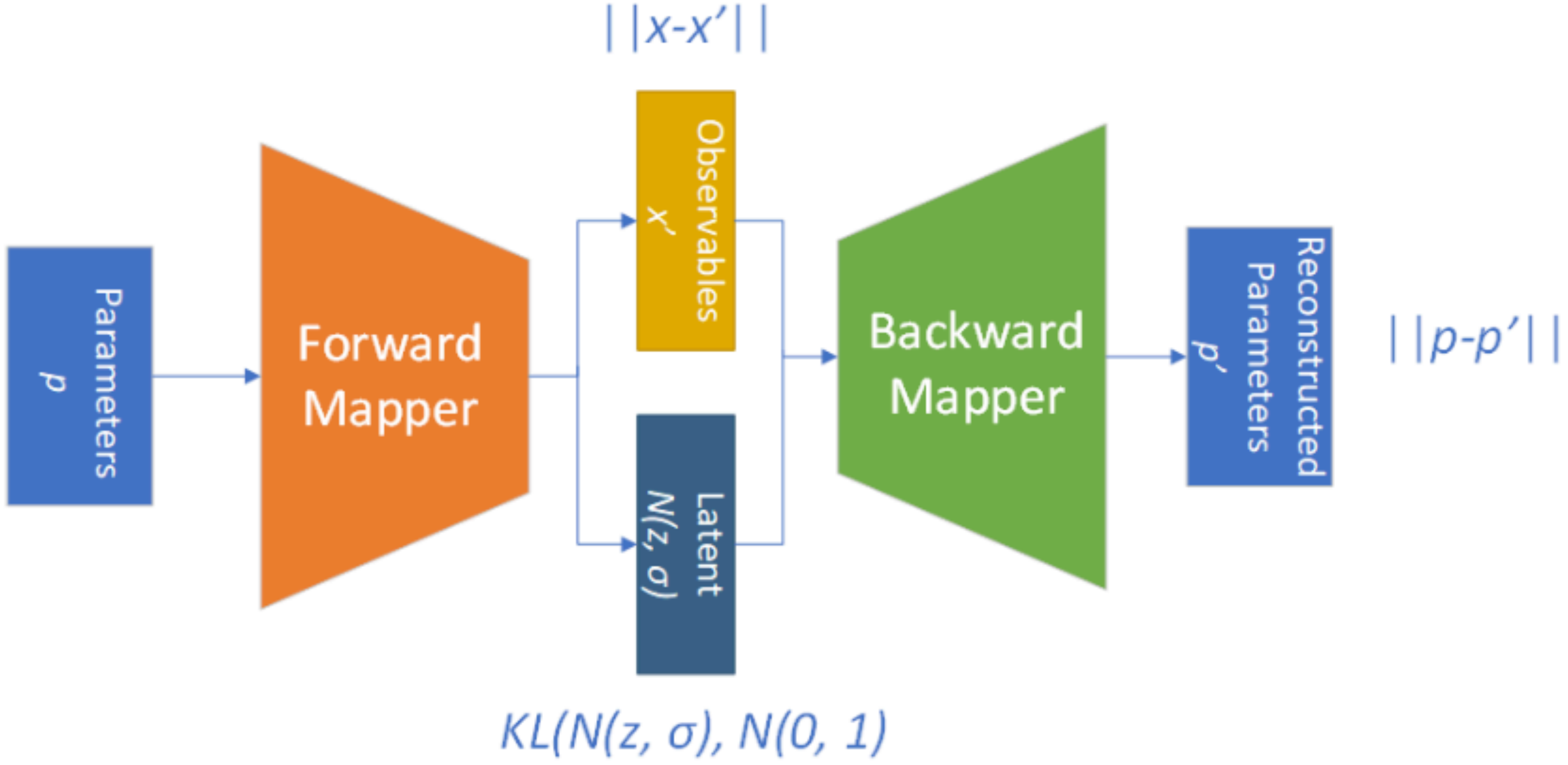
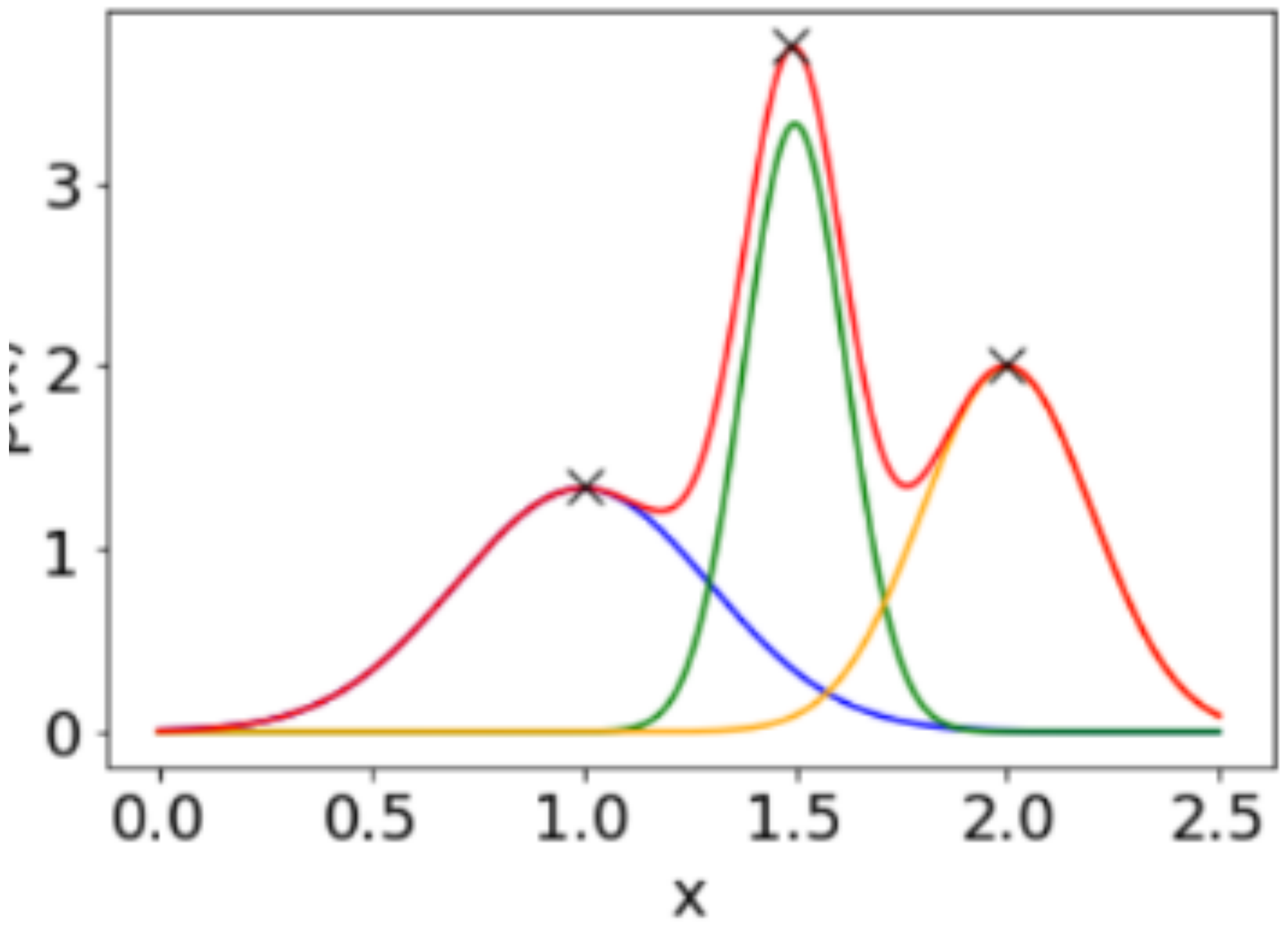
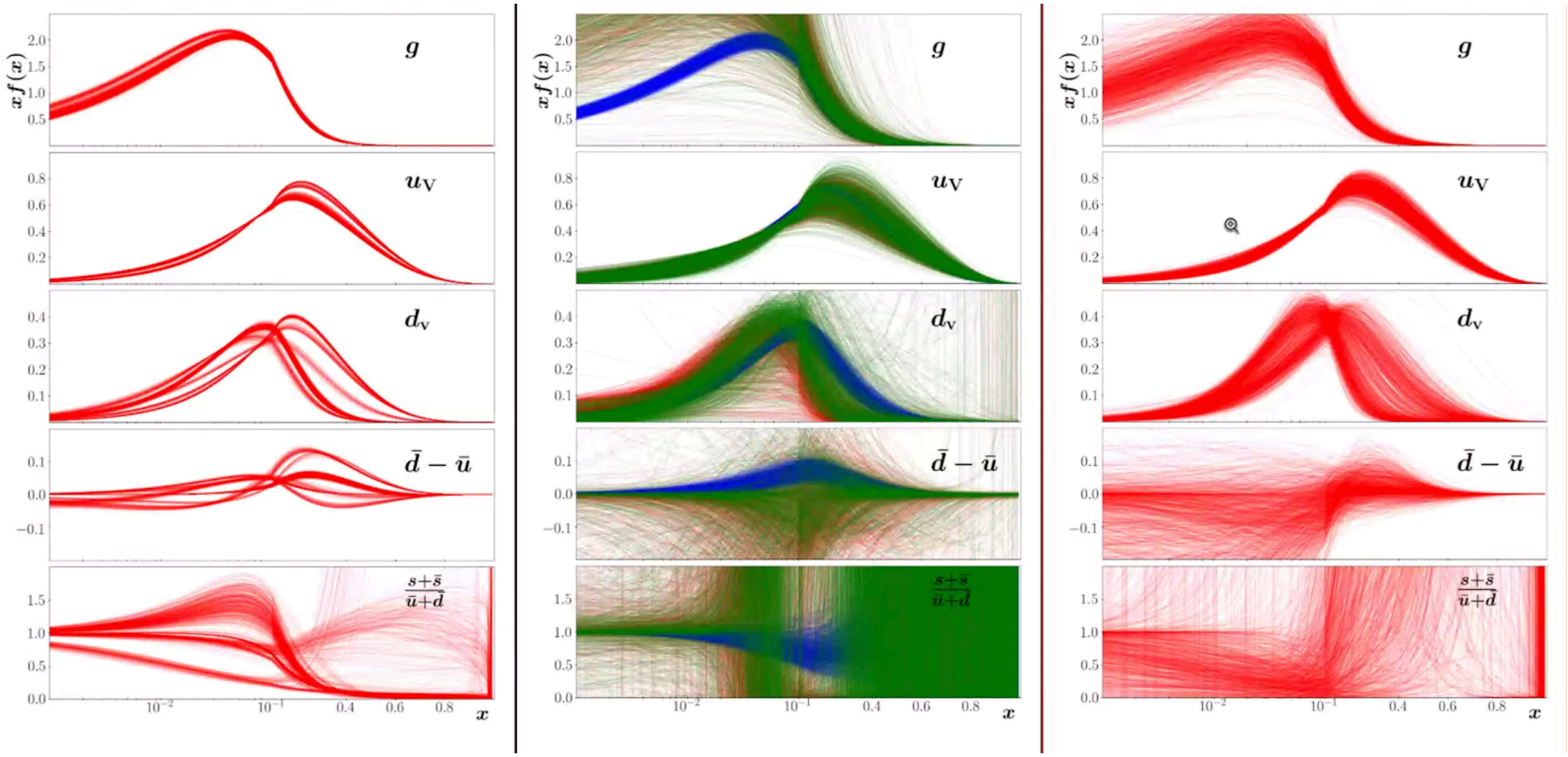


Figure 2: Architecture of the kinematics-independent inverse mapper.



Output Layer Interpretation:

$$p(\mathbf{t}|\mathbf{x}) = \sum_{k=1}^{K} \pi_k(\mathbf{x}) \mathcal{N}\left(\mathbf{t}|\boldsymbol{\mu}_k(\mathbf{x}), \sigma_k^2(\mathbf{x})\right)$$



g
u_V
d_v
d̄ − ū
(s+s̄)/(ū+d̄)
xf(x)
x
10^−2
10^−1
0.4
0.6
0.8

FAST MAPPING TO THEORETICAL PARAMETERS

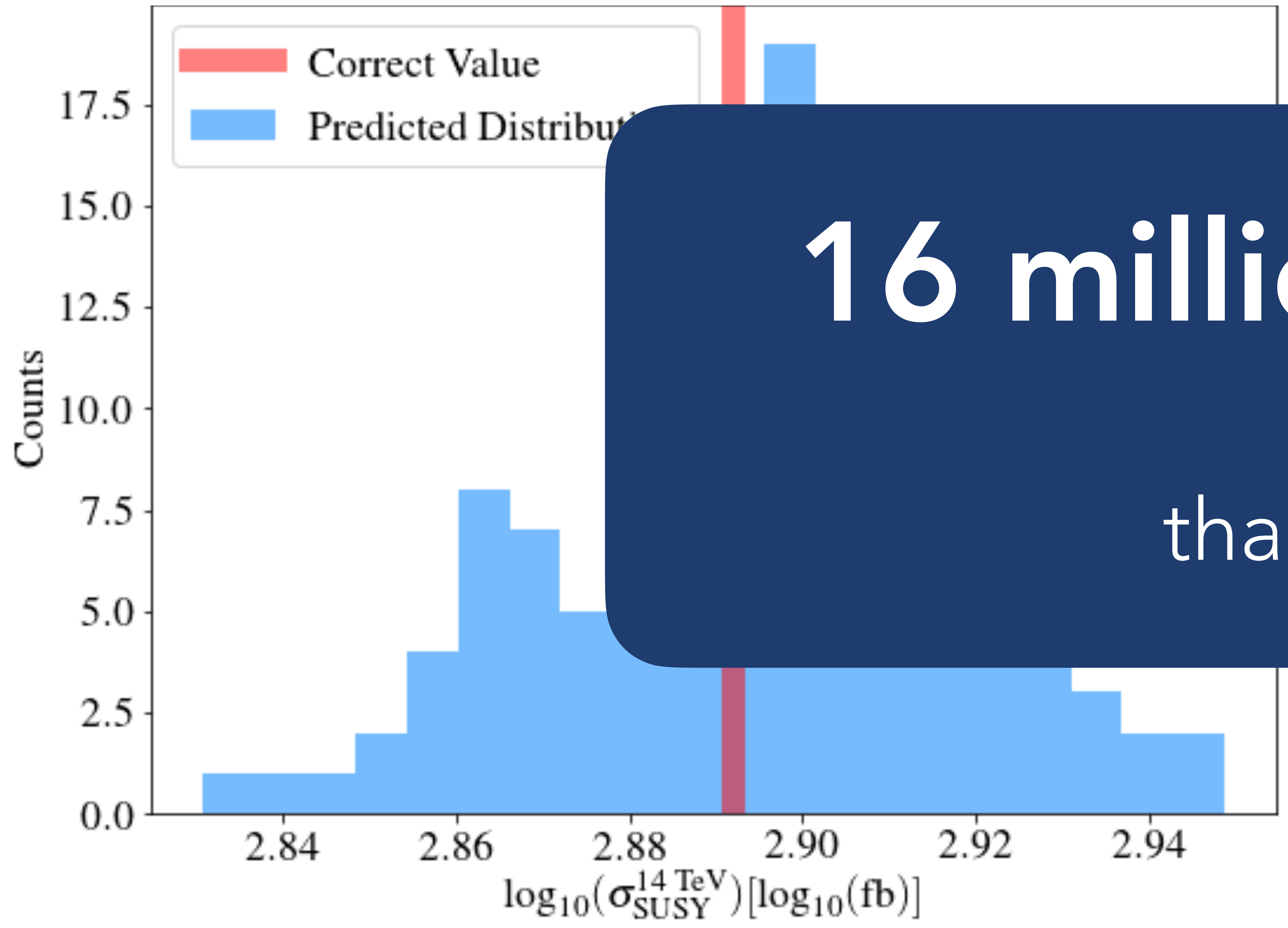
Bayesian Neural Networks

Training — Bayesian inference

Can we make predictions with accurate error estimates?

pMSSM parameters → total SUSY cross section

FAST MAPPING TO THEORETICAL PARAMETERS



16 million times faster

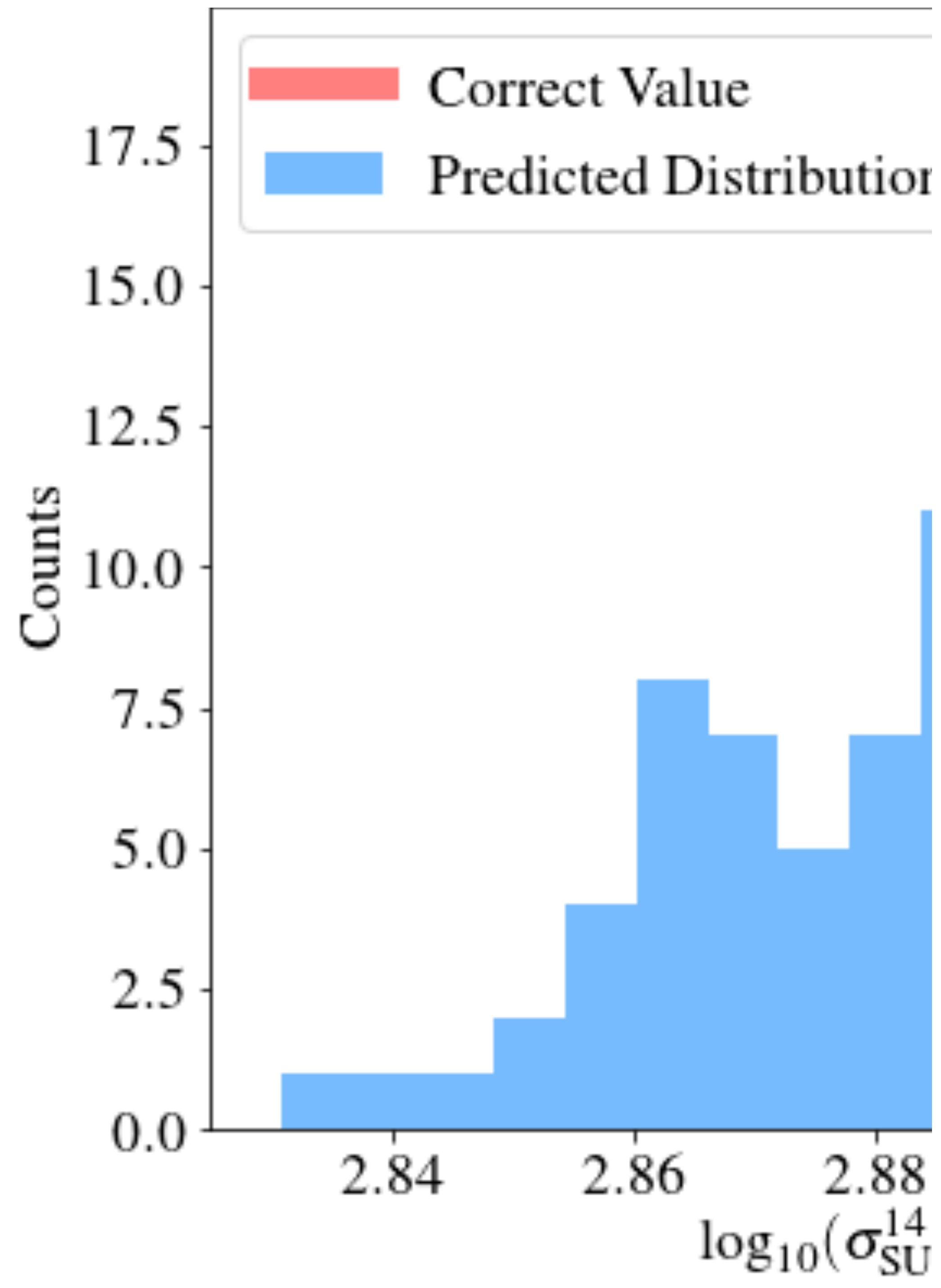
than theory codes!

B.S. Kronheim, M.P. Kuchera, H.B. Prosper, A. Karbo, Bayesian neural networks for fast SUSY predictions, Physics Letters B, Volume 813, 2021, 136041, ISSN 0370-2693, https://doi.org/10.1016/j.physletb.2020.136041.

https://arxiv.org/abs/2009.14393

https://alpha-davidson.github.io/TensorBNN

FAST MAPPING TO THEORETICAL PARAMETERS



Physics Letters B 813 (2021) 136041

Contents lists available at ScienceDirect

Physics Letters B

www.elsevier.com/locate/physletb

Bayesian neural networks for fast SUSY predictions

B.S. Kronheim [a,*], M.P. Kuchera [a], H.B. Prosper [b], A. Karbo [a]

[a] Department of Physics, Davidson College, Davidson, NC 28035, USA
[b] Department of Physics, Florida State University, Tallahassee, FL 32306, USA

Computer Physics Communications 270 (2022) 108168

Contents lists available at ScienceDirect

Computer Physics Communications

www.elsevier.com/locate/cpc

TensorBNN: Bayesian inference for neural networks using TensorFlow ☆

B.S. Kronheim [a,*,1], M.P. Kuchera [a], H.B. Prosper [b]

[a] Department of Physics, Davidson College, Davidson, NC 28036, United States of America
[b] Department of Physics, Florida State University, Tallahassee, FL 32306, United States of America

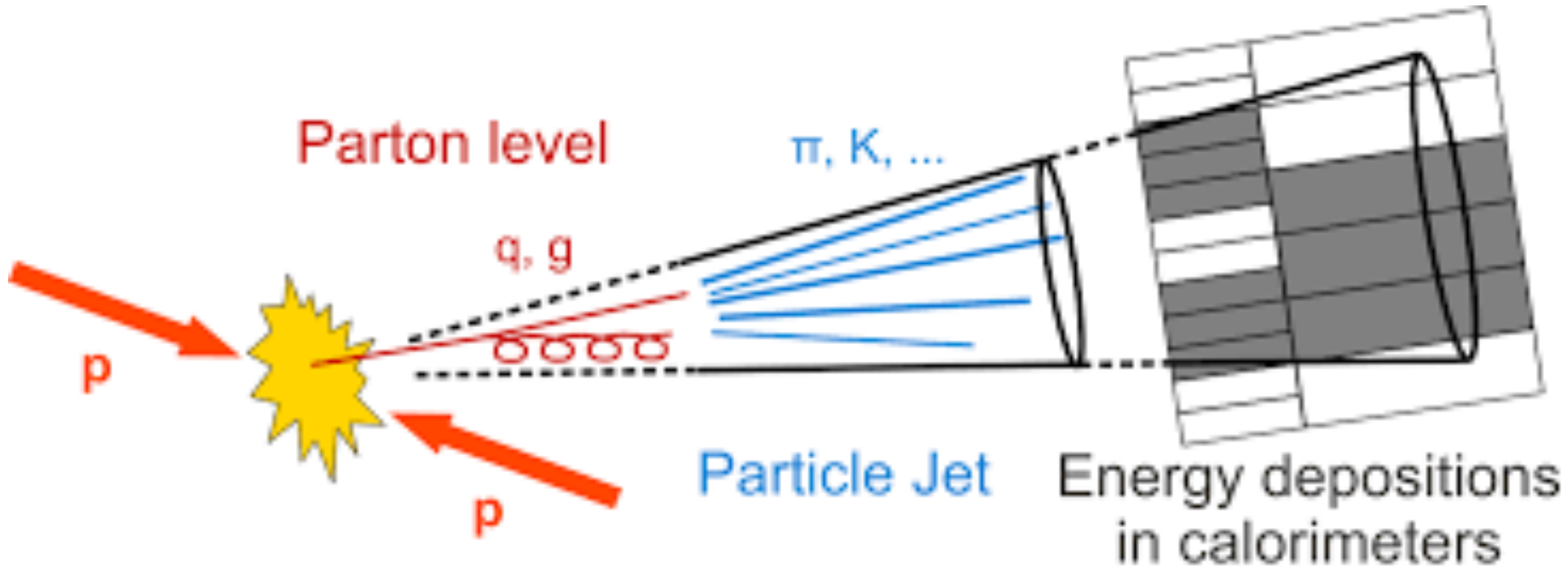
B.S. Kronheim, M.P. Kuchera, H.B. Prosper,
predictions, Physics Letters B, Volume 813
10.1016/j.physletb.2020.136041.

https://arxiv.org/abs/2009.14393

https://alpha-davidson.github.io/TensorBNN

Application 2: How can make *accurate* predictions for stochastic processes?

JET SIMULATION AND CORRECTION



DATASET: CMS Collaboration (2019). Simulated dataset QCD_Pt-15to7000_TuneCUETP8M1_Flat_13TeV_pythia8 in MINIAODSIM format for 2016 collision data. CERN Open Data Portal. DOI:10.7483/OPENDATA.CMS.J52Q.4T4E

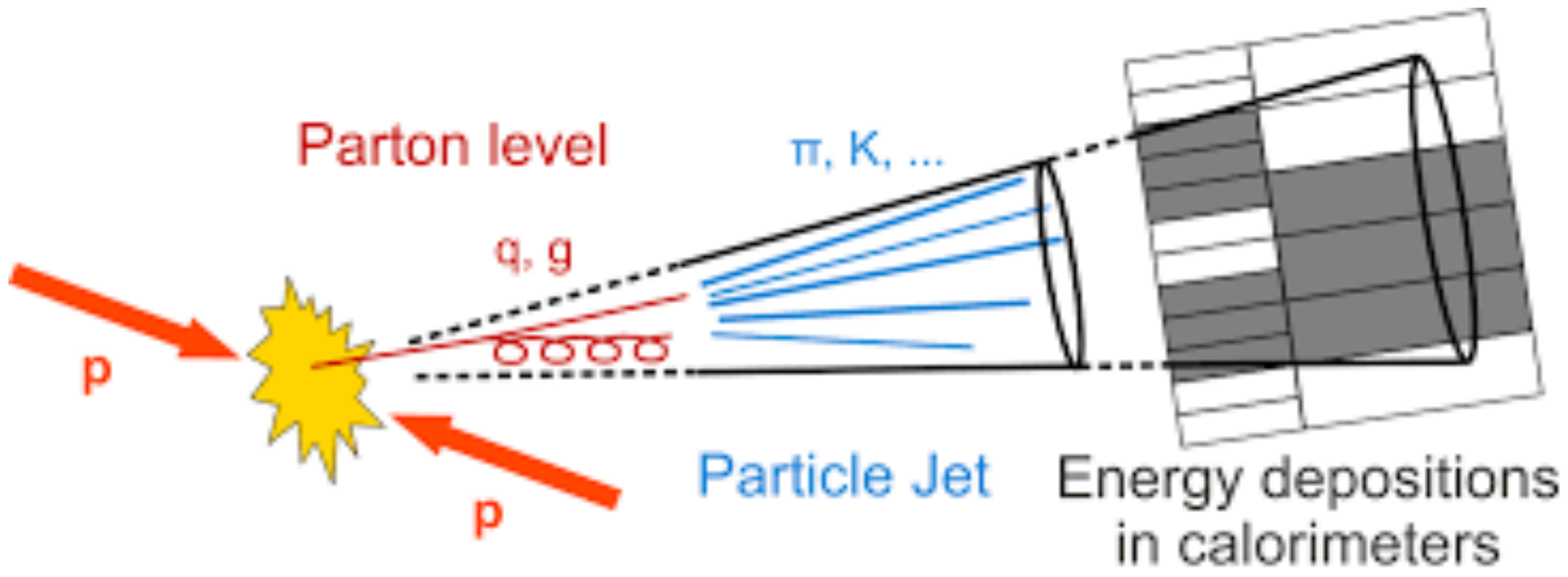
JET SIMULATION AND CORRECTION

NEED FOR DISTRIBUTION PREDICTIONS

JET SIMULATION AND CORRECTION

J. BLUE, ET.AL., CHEP '21
EPJ WOC **251**, 03055 (2021) HTTPS://DOI.ORG/10.1051/EPJCONF/202125103055

EXISTING METHODS



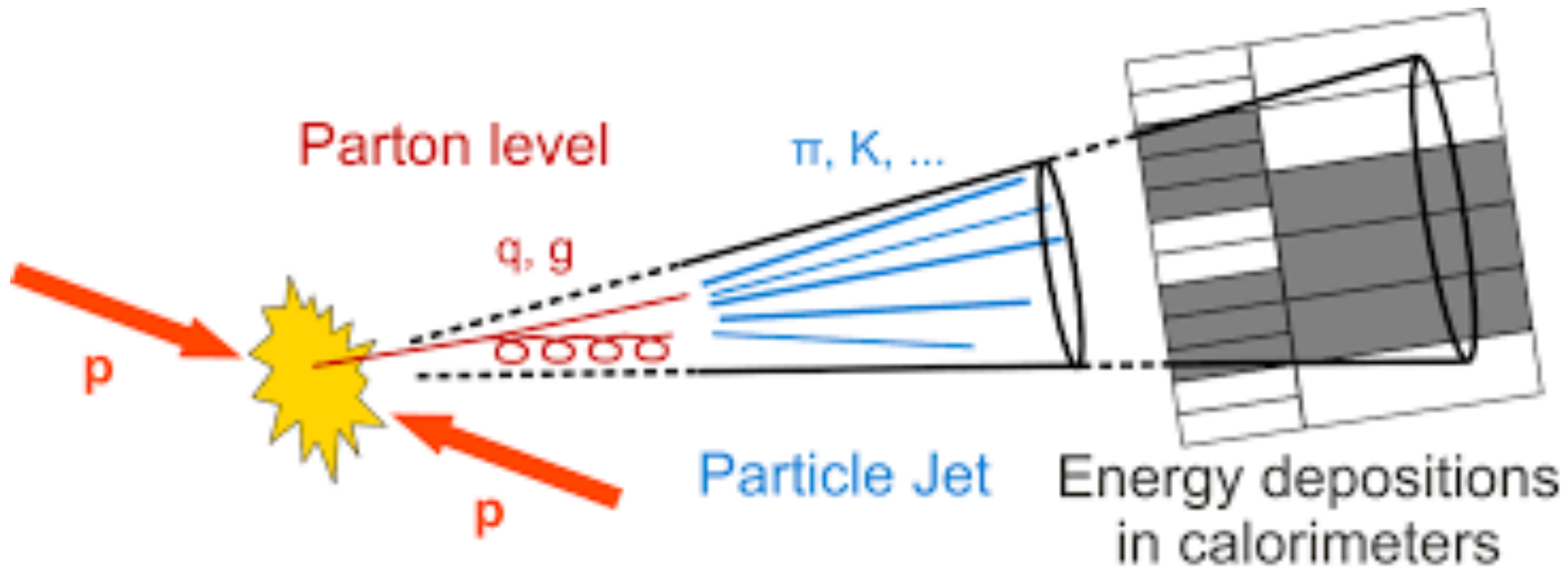
(conditional) generative adversarial networks

arXiv:1912.00477
arXiv:1807.01954
arXiv:1805.00850
arXiv:1712.10321

normalizing flows

arXiv:1904.12072
arXiv:2001.05486
arXiv:2001.10028
arXiv:2012.09873
arXiv:2106.05285

EXISTING METHODS



(conditional) generative adversarial networks

arXiv:1912.00477

arXiv:1807.01954

arXiv:1805.00850

arXiv:1712.10321

How to GAN away Detector Effects

Marco Bellagente[1], Anja Butter[1], Gregor Kasieczka[2], Tilman Plehn[1], and Ramon Winterhalder[1]

1 Institut für Theoretische Physik, Universität Heidelberg, Germany
2 Institut für Experimentalphysik, Universität Hamburg, Germany
bellagente@thphys.uni-heidelberg.de

Precise simulation of electromagnetic calorimeter showers using a Wasserstein Generative Adversarial Network

Martin Erdmann[a] Jonas Glombitza[a] Thorben Quast[a,b]

[a] *III. Physikalisches Institut A, Rheinisch Westfälische Technische Hochschule, Aachen, Germany*
[b] *EP-LCD, CERN, Geneva, Switzerland*

Fast and accurate simulation of particle detectors using generative adversarial networks

Pasquale Musella · Francesco Pandolfi

CaloGAN: Simulating 3D High Energy Particle Showers in Multi-Layer Electromagnetic Calorimeters with Generative Adversarial Networks

Michela Paganini,[1, 2, *] Luke de Oliveira,[2, †] and Benjamin Nachman[2, ‡]

[1] *Department of Physics, Yale University, New Haven, CT 06520, USA*
[2] *Lawrence Berkeley National Laboratory, Berkeley, CA, 94720, USA*

(Dated: January 1, 2018)

Flow-based generative models for Markov chain Monte Carlo in lattice field theory

M. S. Albergo,[1,2,3] G. Kanwar,[4] and P. E. Shanahan[4,1]

[1] *Perimeter Institute for Theoretical Physics, Waterloo, Ontario N2L 2Y5, Canada*
[2] *Cavendish Laboratories, University of Cambridge, Cambridge CB3 0HE, U.K.*
[3] *University of Waterloo, Waterloo, Ontario N2L 3G1, Canada*
[4] *Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.*

i-flow: High-dimensional Integration and Sampling with Normalizing Flows

Christina Gao[1], Joshua Isaacson[1], and Claudius Krause[1]

[1] Theoretical Physics Department, Fermi National Accelerator Laboratory, Batavia, IL, 60510, USA

Event Generation with Normalizing Flows

Christina Gao,[1] Stefan Höche,[1] Joshua Isaacson,[1] Claudius Krause,[1] and Holger Schulz[2]

[1] *Fermi National Accelerator Laboratory, Batavia, IL, 60510, USA*
[2] *Department of Physics, University of Cincinnati, Cincinnati, OH 45219, USA*

Measuring QCD Splittings with Invertible Networks

Sebastian Bieringer[1], Anja Butter[1], Theo Heimel[1], Stefan Höche[2], Ullrich Köthe[3], Tilman Plehn[1], and Stefan T. Radev[4]

1 Institut für Theoretische Physik, Universität Heidelberg, Germany
2 Fermi National Accelerator Laboratory, Batavia, IL, USA
3 Heidelberg Collaboratory for Image Processing, Universität Heidelberg, Germany
4 Psychologisches Institut, Universität Heidelberg, Germany
heimel@thphys.uni-heidelberg.de

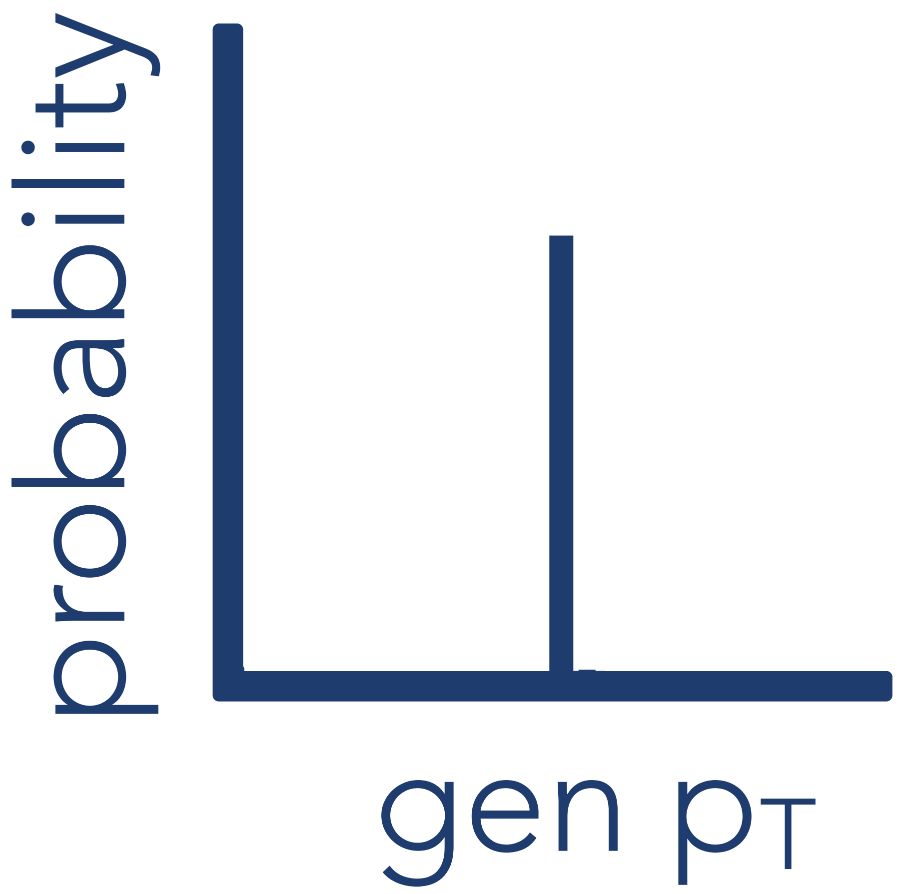
CaloFlow: Fast and Accurate Generation of Calorimeter Showers with Normalizing Flows

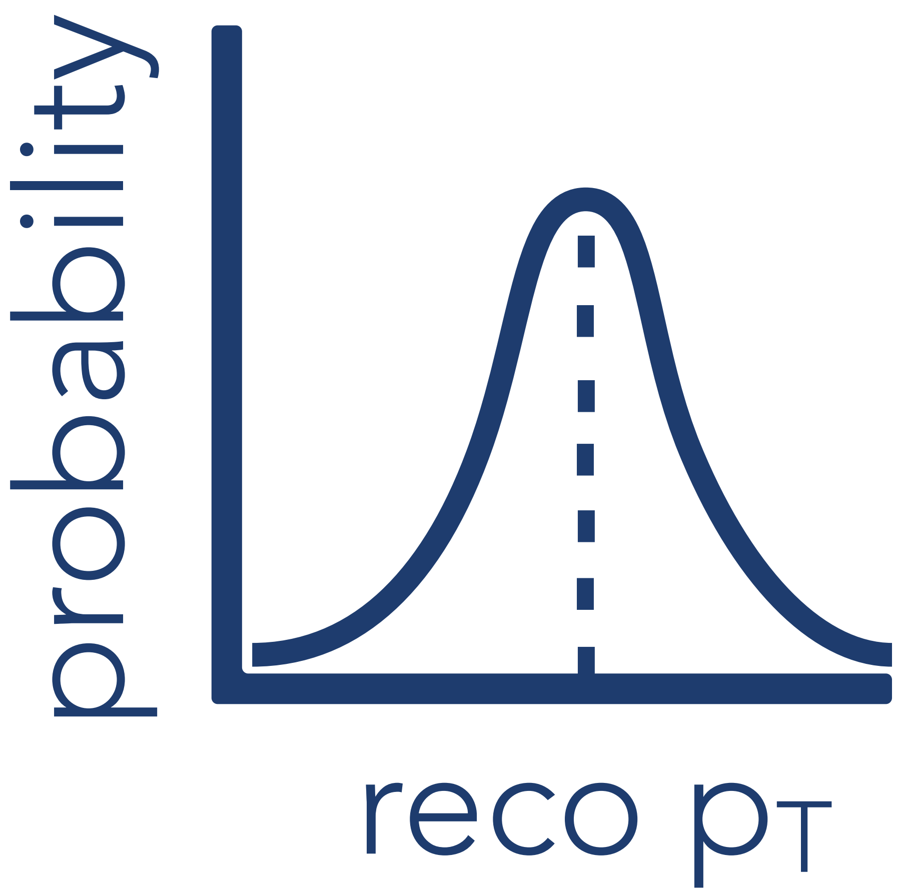
---

Claudius Krause and David Shih

NHETC, Dept. of Physics and Astronomy, Rutgers University, Piscataway, NJ 08854, USA

E-mail: Claudius.Krause@rutgers.edu, shih@physics.rutgers.edu

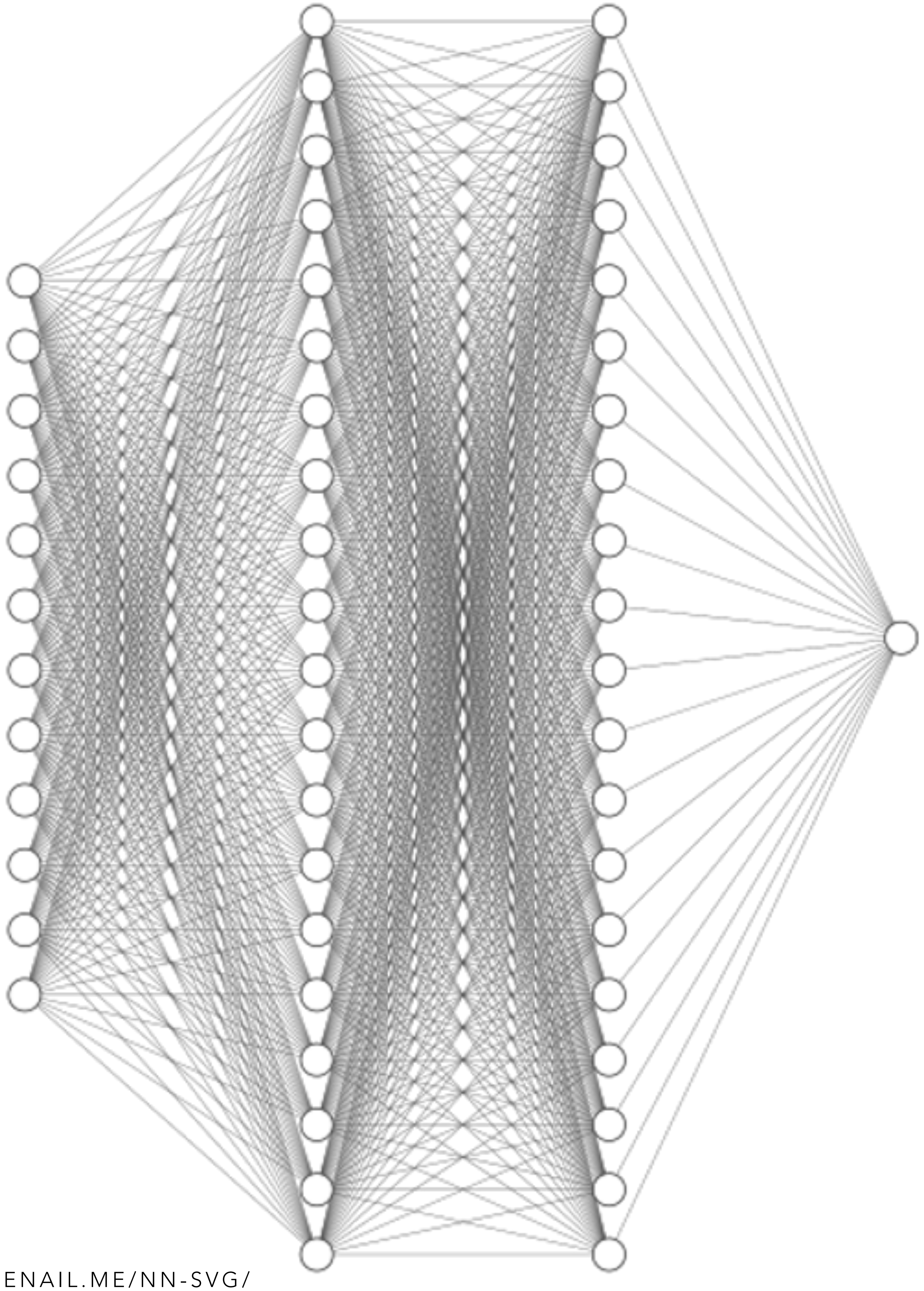




normalizing flows

arXiv:1904.12072
arXiv:2001.05486
arXiv:2001.10028
arXiv:2012.09873
arXiv:2106.05285

IMPLICIT QUANTILE NETWORKS ARCHITECTURE

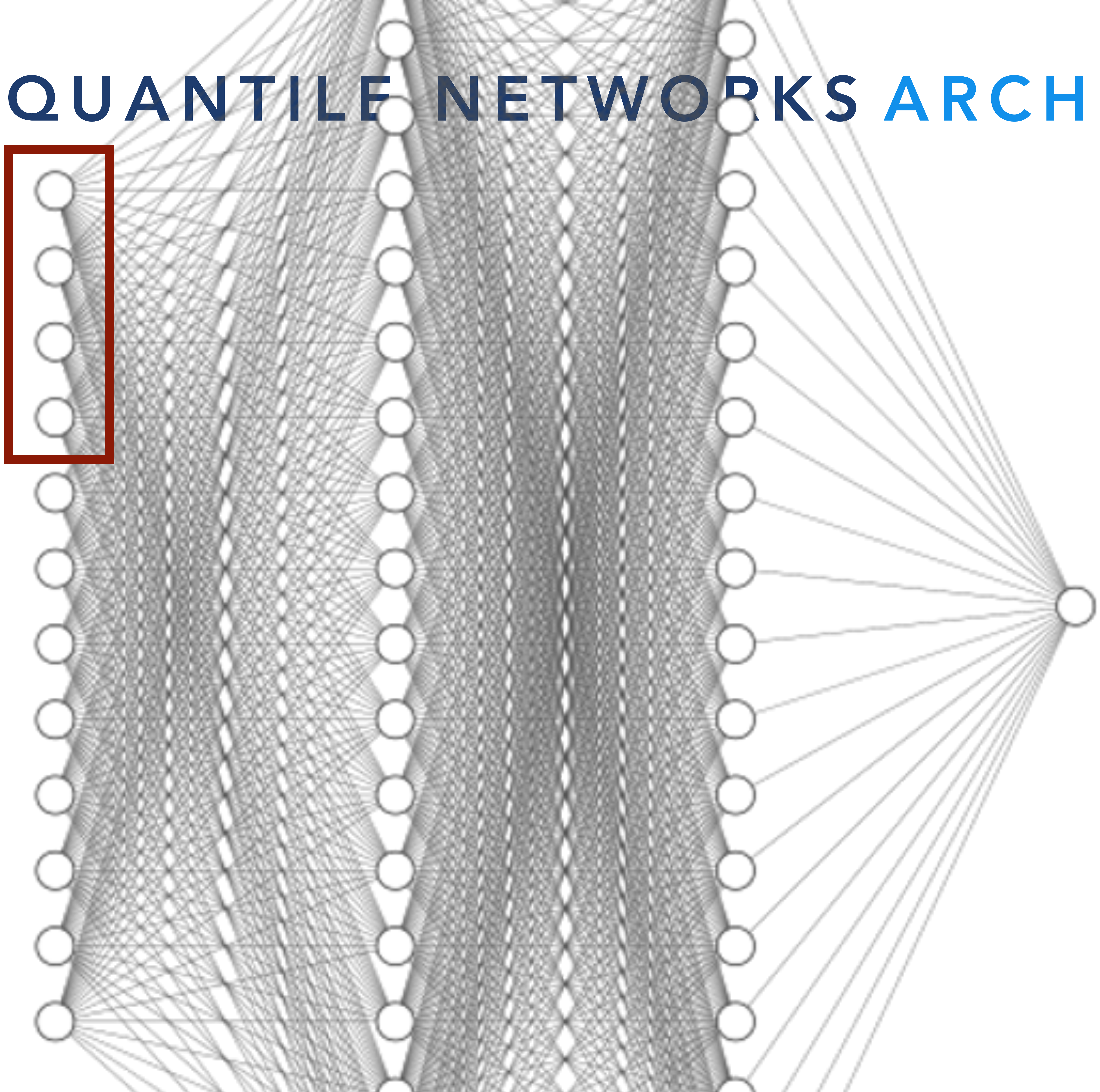


$$(p_T, \eta, \phi, m) \rightarrow (p'_T, \eta', \phi', m')$$

IMAGE CREDIT: HTTPS://ALEXLENAIL.ME/NN-SVG/

IMPLICIT QUANTILE NETWORKS ARCHITECTURE

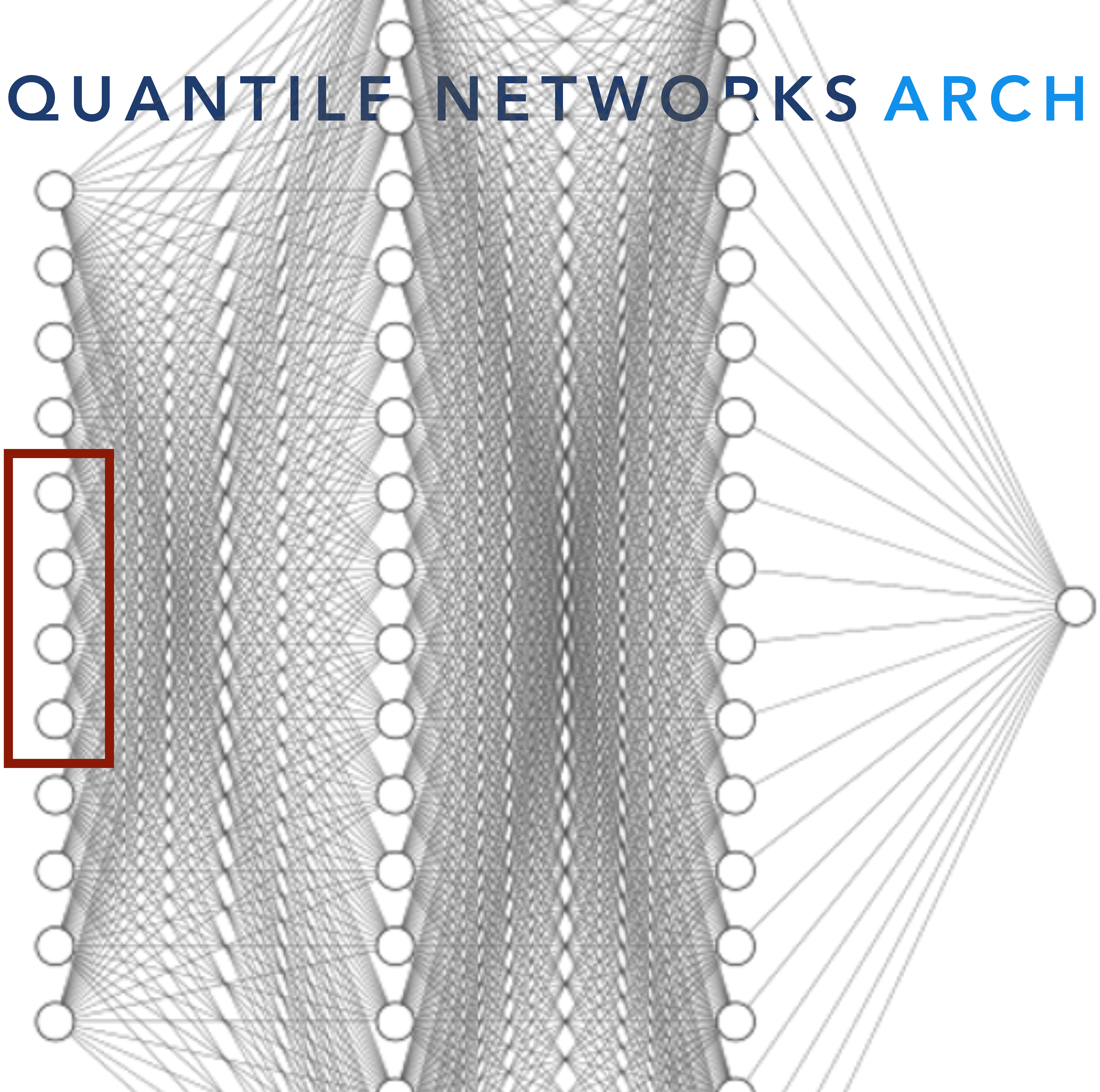
(p_T, η, ϕ, m)



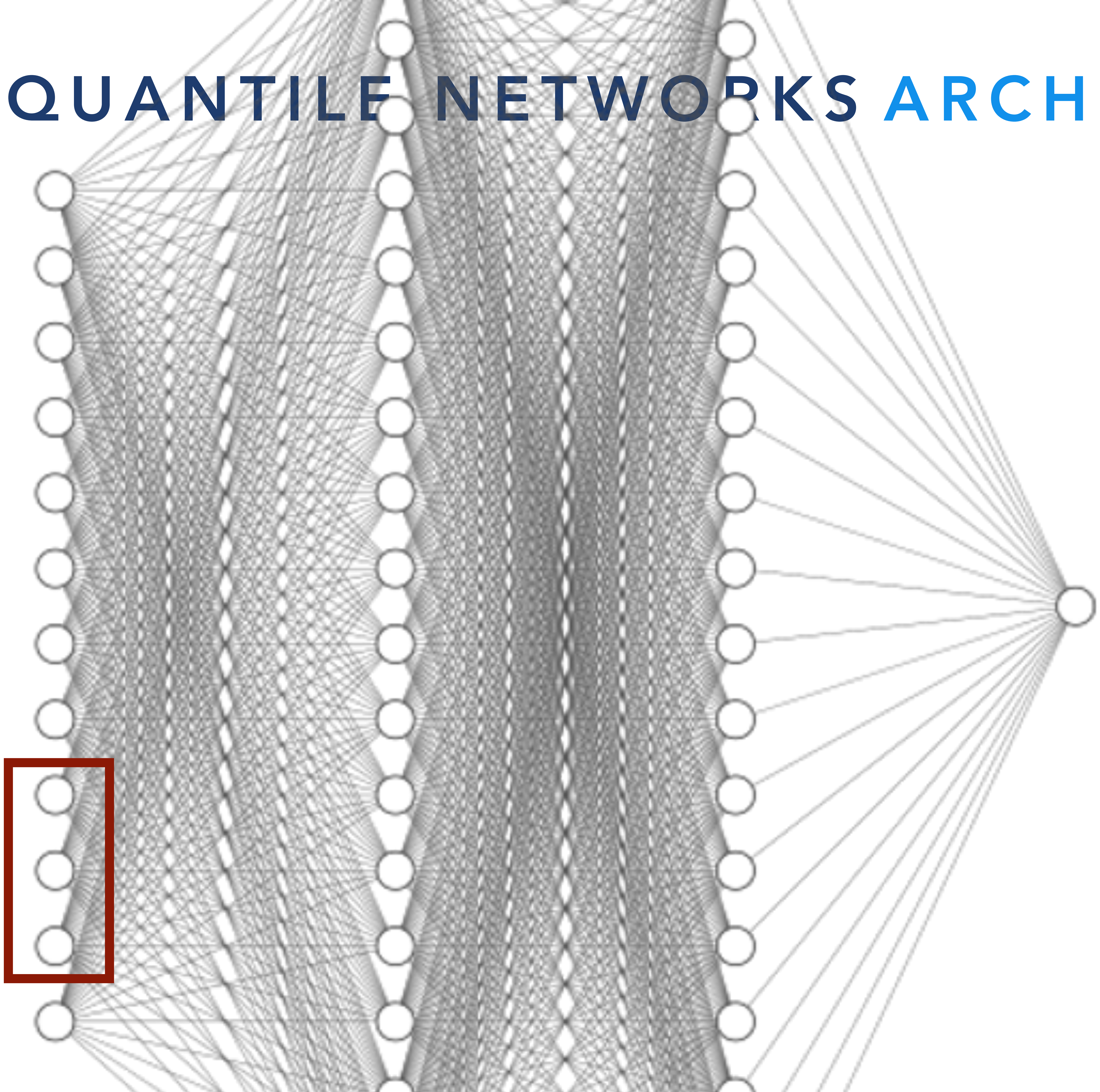
IMPLICIT QUANTILE NETWORKS ARCHITECTURE

(p_T, η, ϕ, m)

[0,0,1,0]

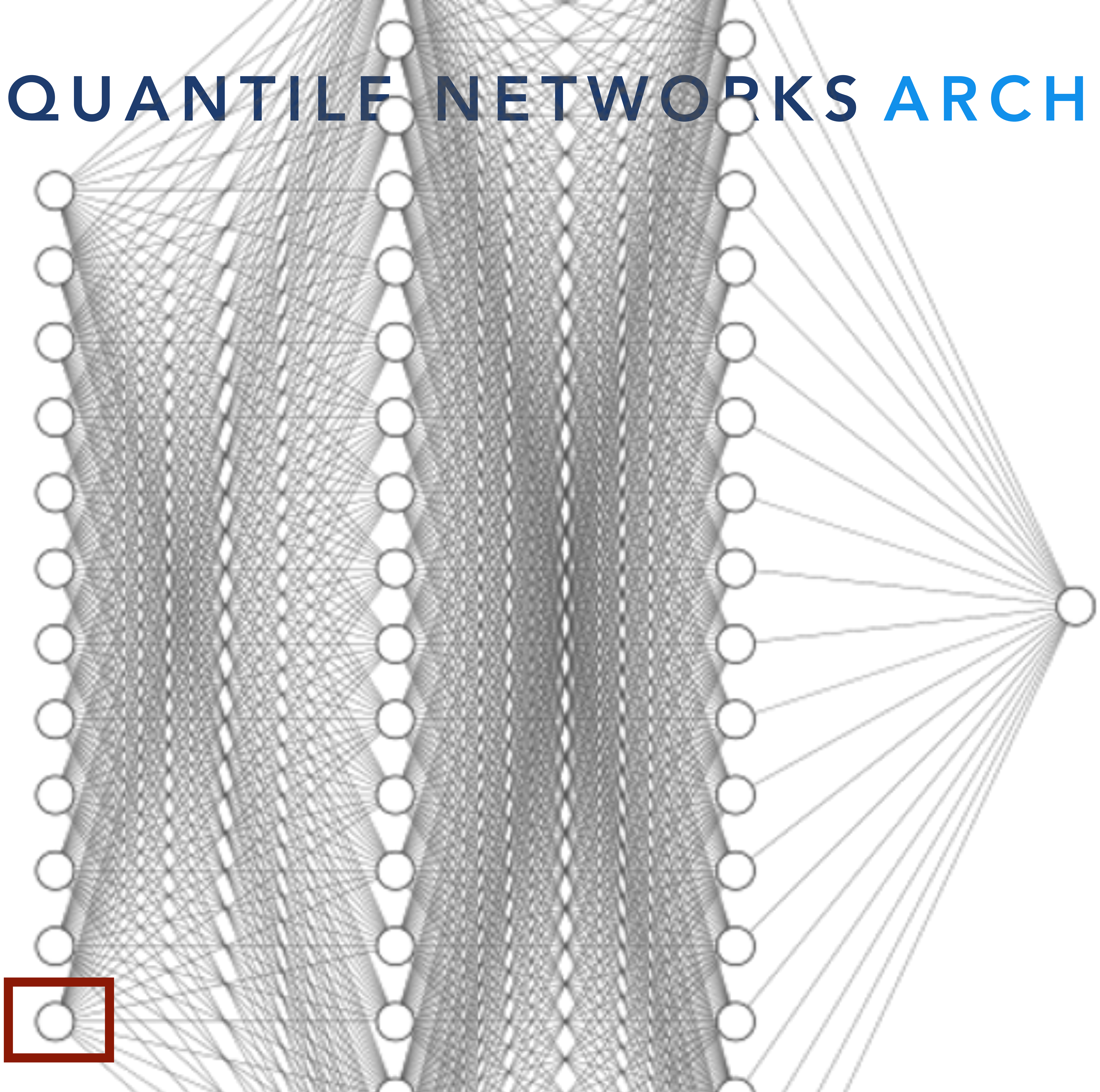


IMPLICIT QUANTILE NETWORKS ARCHITECTURE



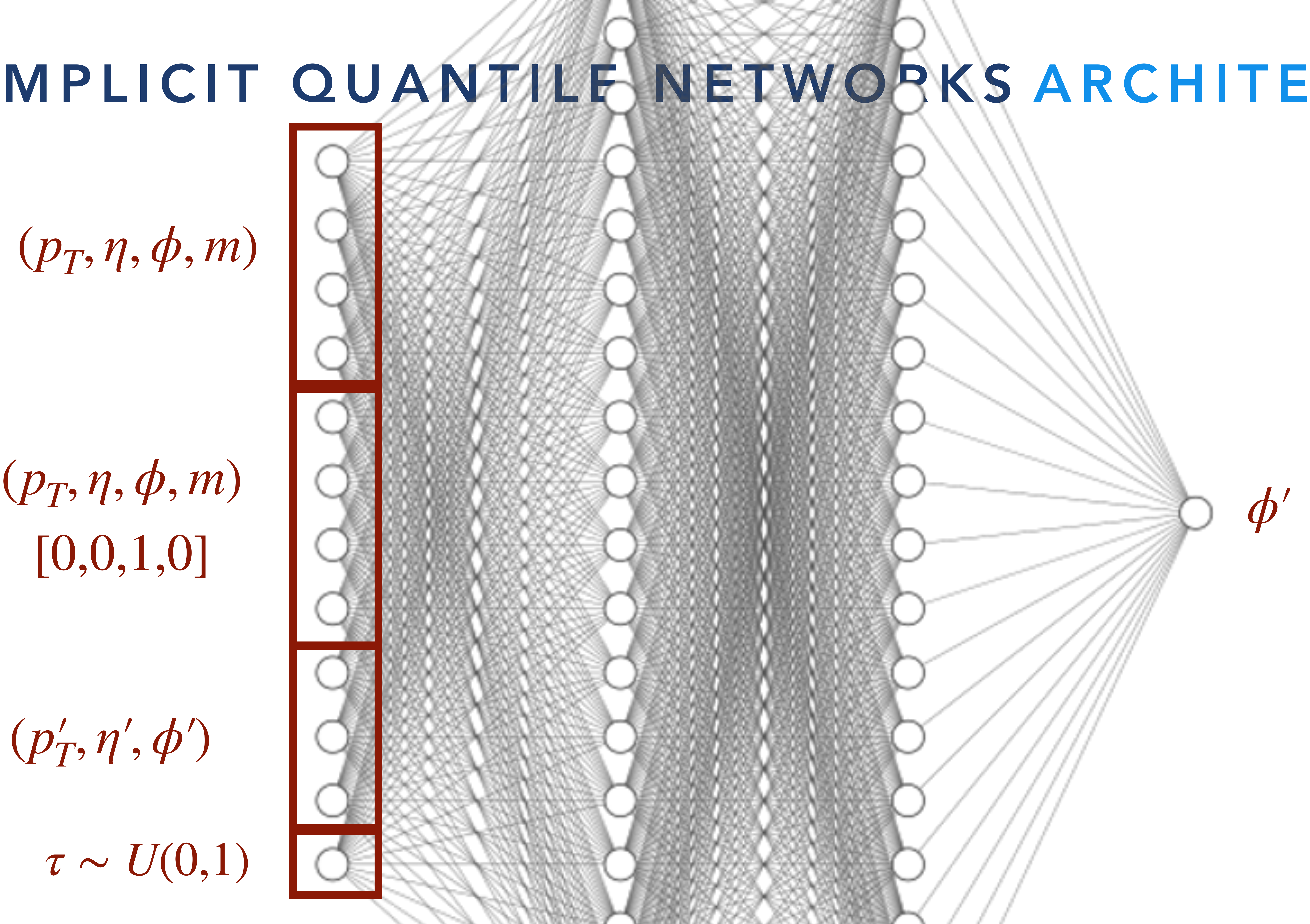
(p_T', η', ϕ')

IMPLICIT QUANTILE NETWORKS ARCHITECTURE

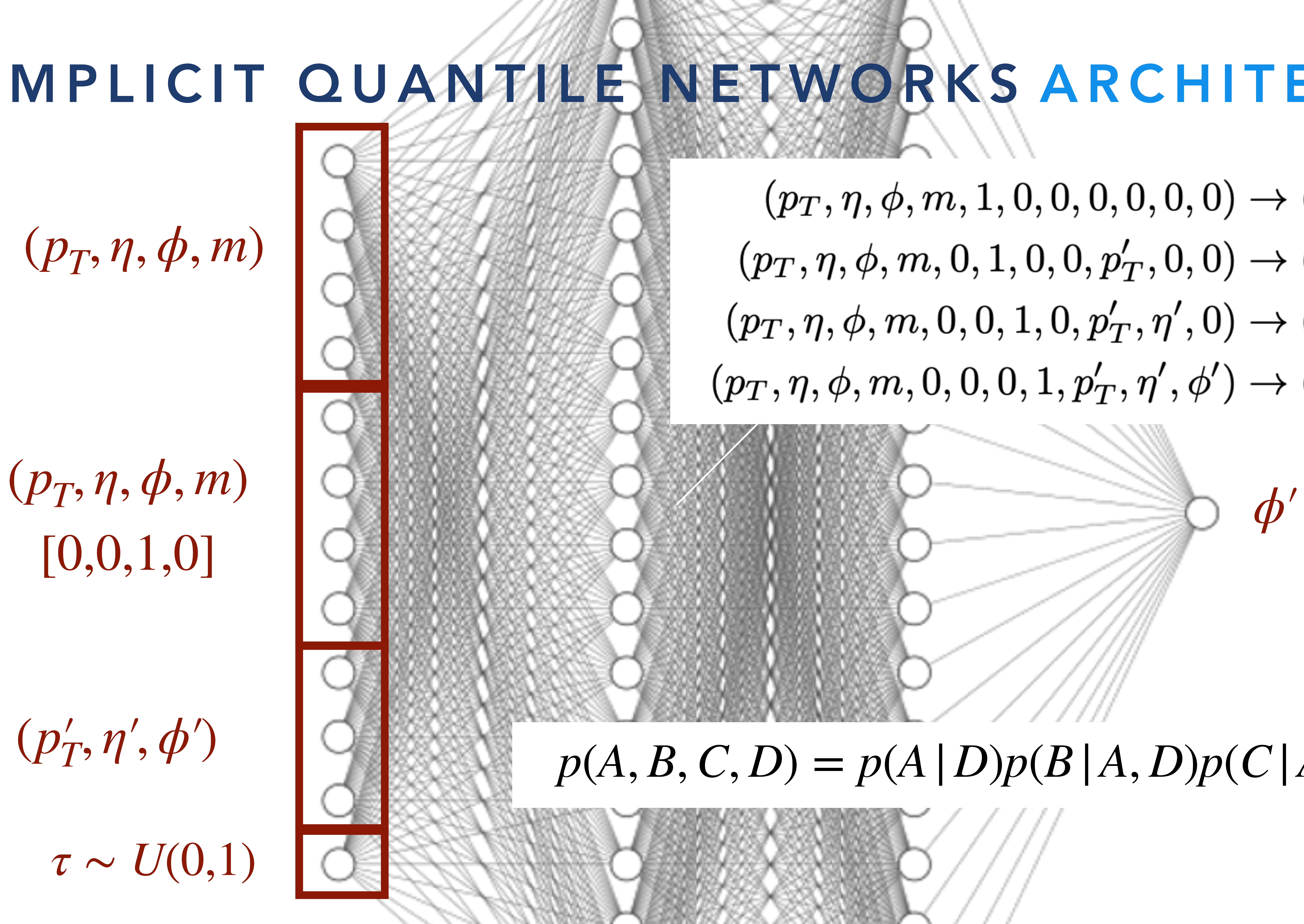


$\tau \sim U(0,1)$

IMPLICIT QUANTILE NETWORKS ARCHITECTURE



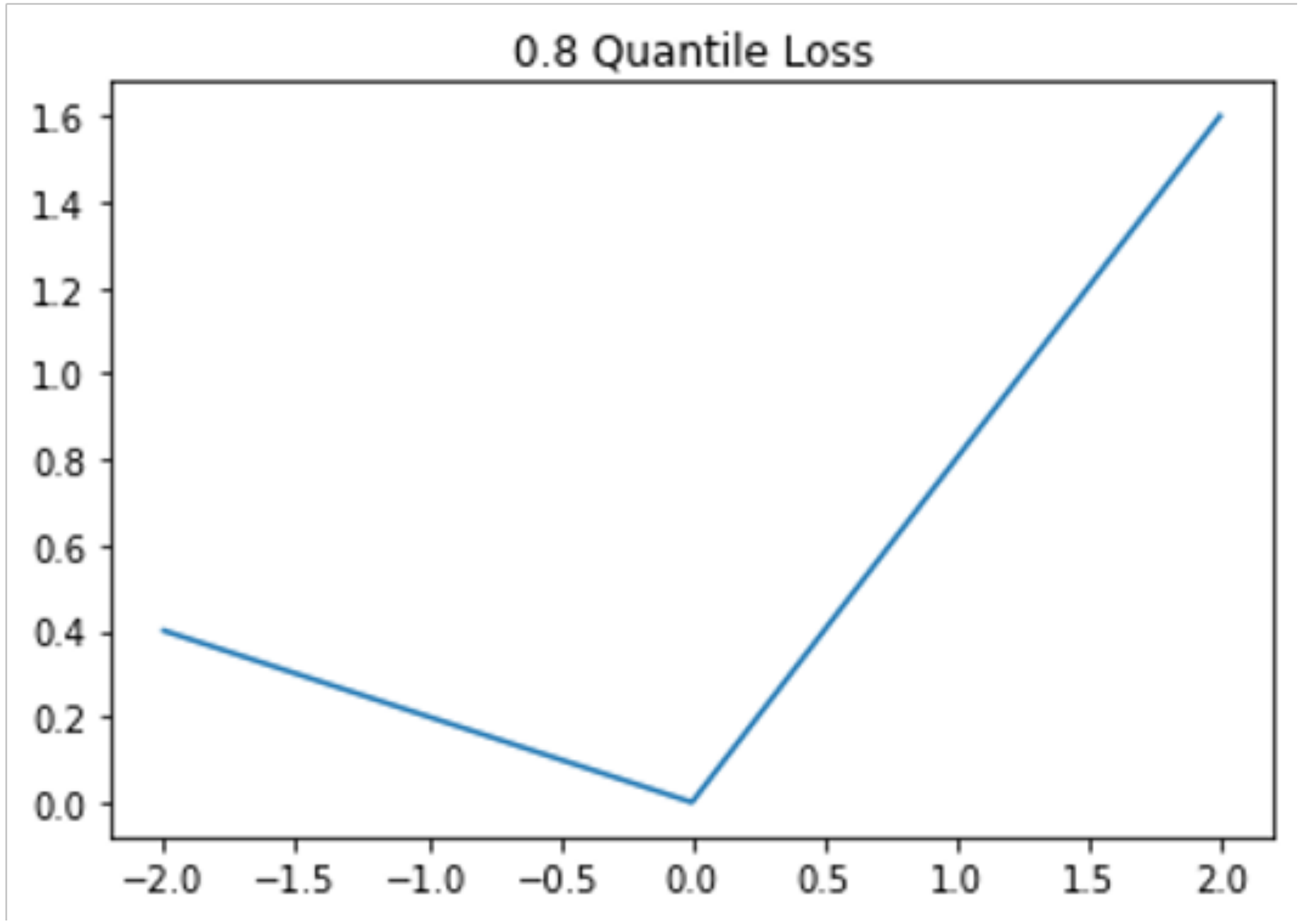
IMPLICIT QUANTILE NETWORKS ARCHITECTURE



$$
\begin{aligned}
(p_T, \eta, \phi, m, 1, 0, 0, 0, 0, 0, 0) &\rightarrow (p'_T), \\
(p_T, \eta, \phi, m, 0, 1, 0, 0, p'_T, 0, 0) &\rightarrow (\eta'), \\
(p_T, \eta, \phi, m, 0, 0, 1, 0, p'_T, \eta', 0) &\rightarrow (\phi'), \\
(p_T, \eta, \phi, m, 0, 0, 0, 1, p'_T, \eta', \phi') &\rightarrow (m'),
\end{aligned}
$$

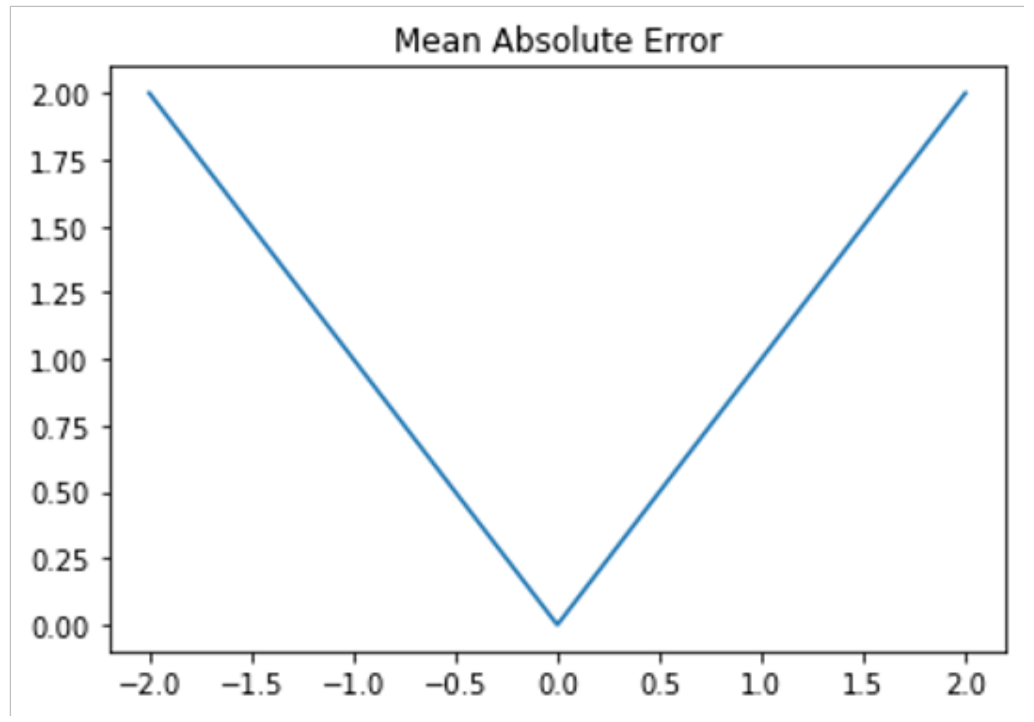
$$p(A, B, C, D) = p(A \,|\, D)\, p(B \,|\, A, D)\, p(C \,|\, A, B, D)$$

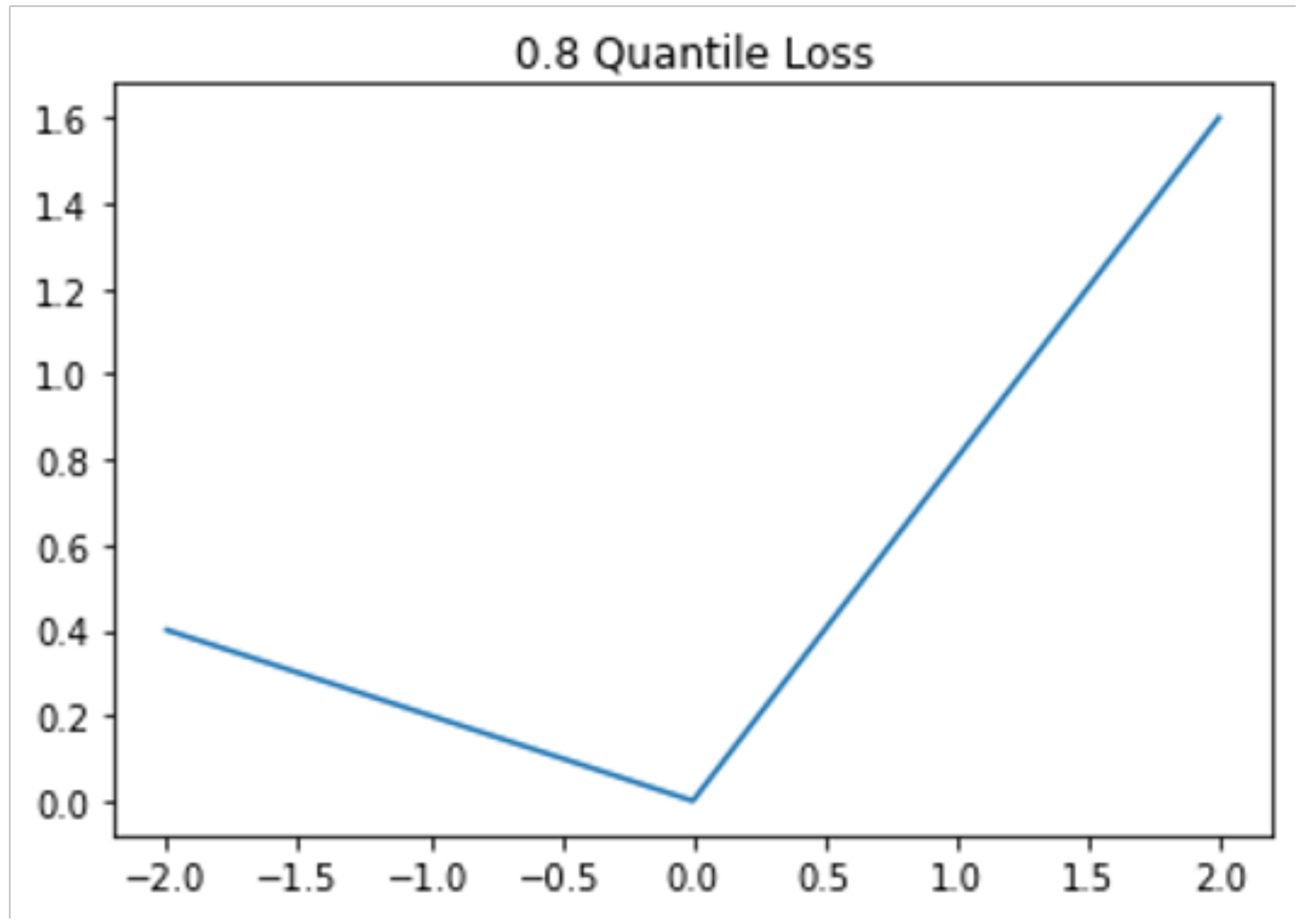
IMPLICIT QUANTILE NETWORKS LOSS FUNCTION

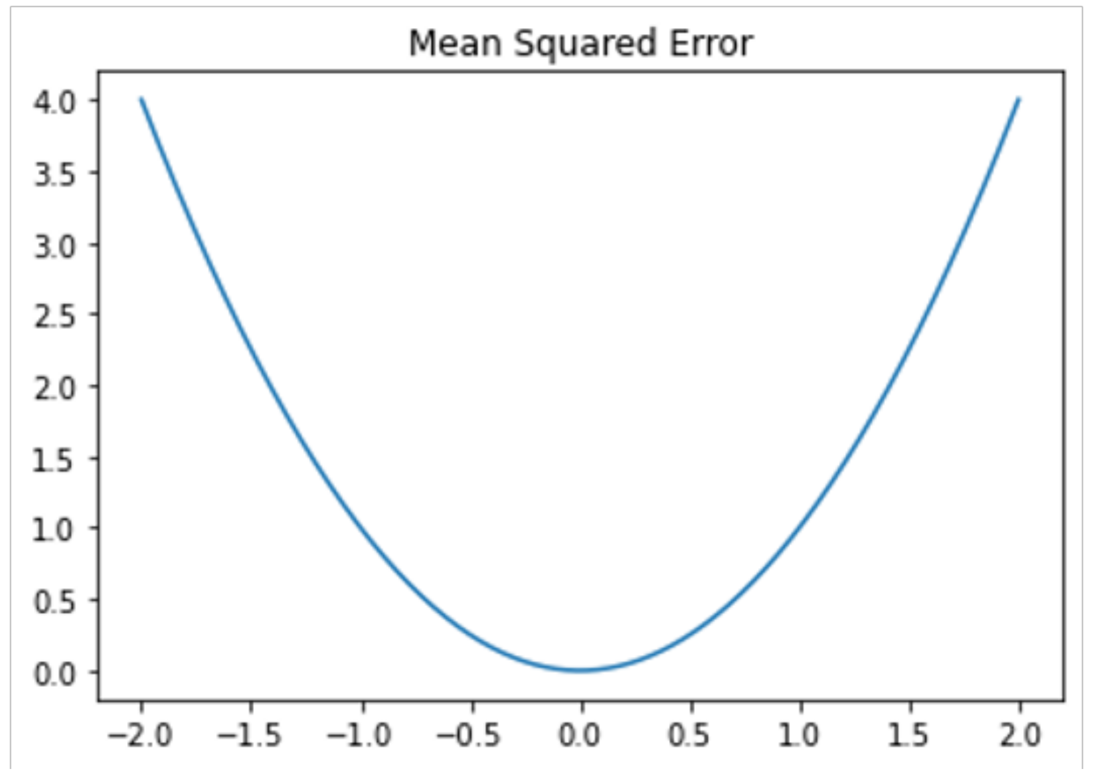


$$\mathcal{L}(f, x, y, \tau) = \begin{cases} \tau(y - f(x, \tau)) & y \geq f(x, \tau) \\ (\tau - 1)(y - f(x, \tau)) & y < f(x, \tau) \end{cases}$$

IMPLICIT QUANTILE NETWORKS LOSS FUNCTION

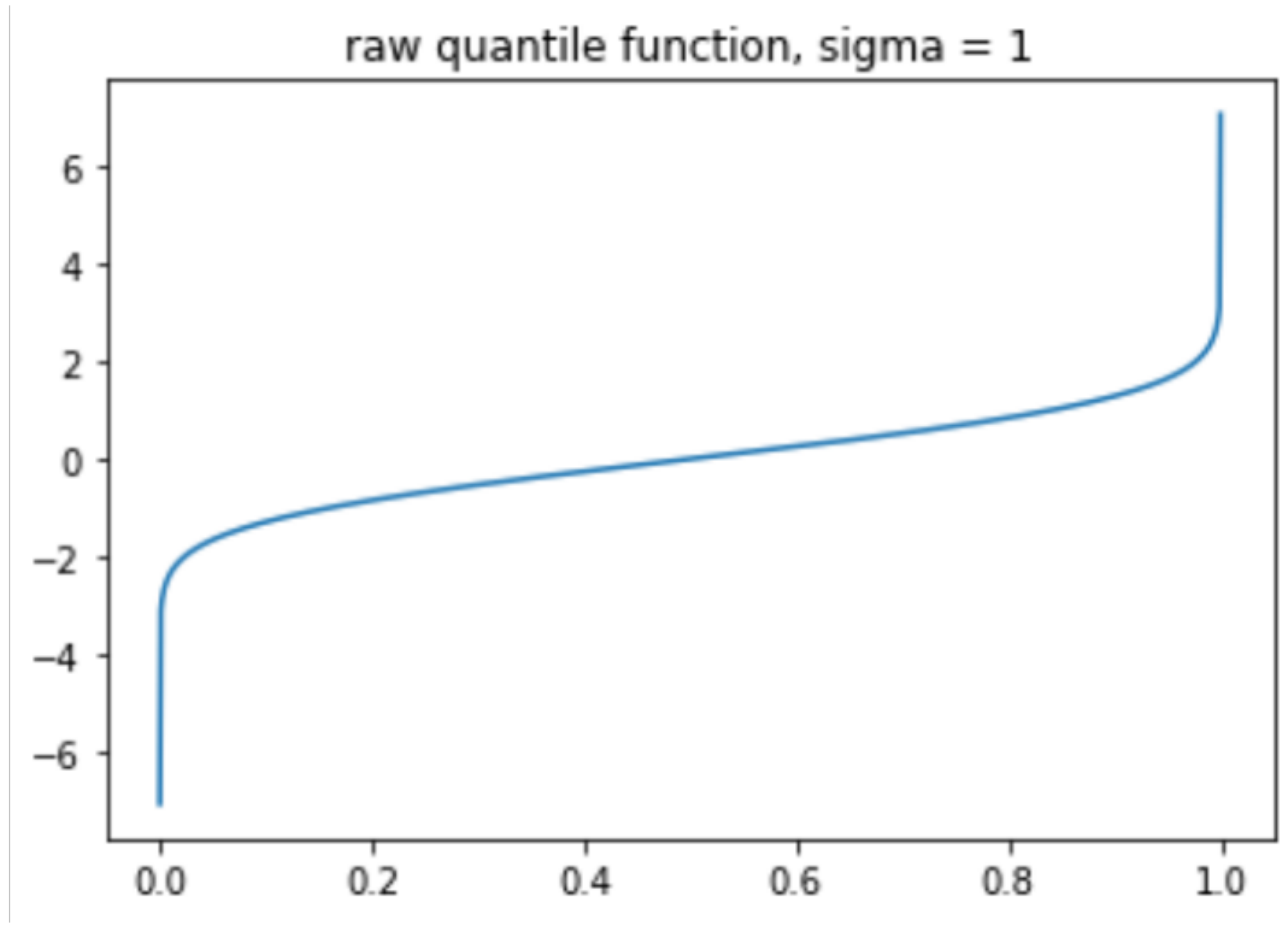


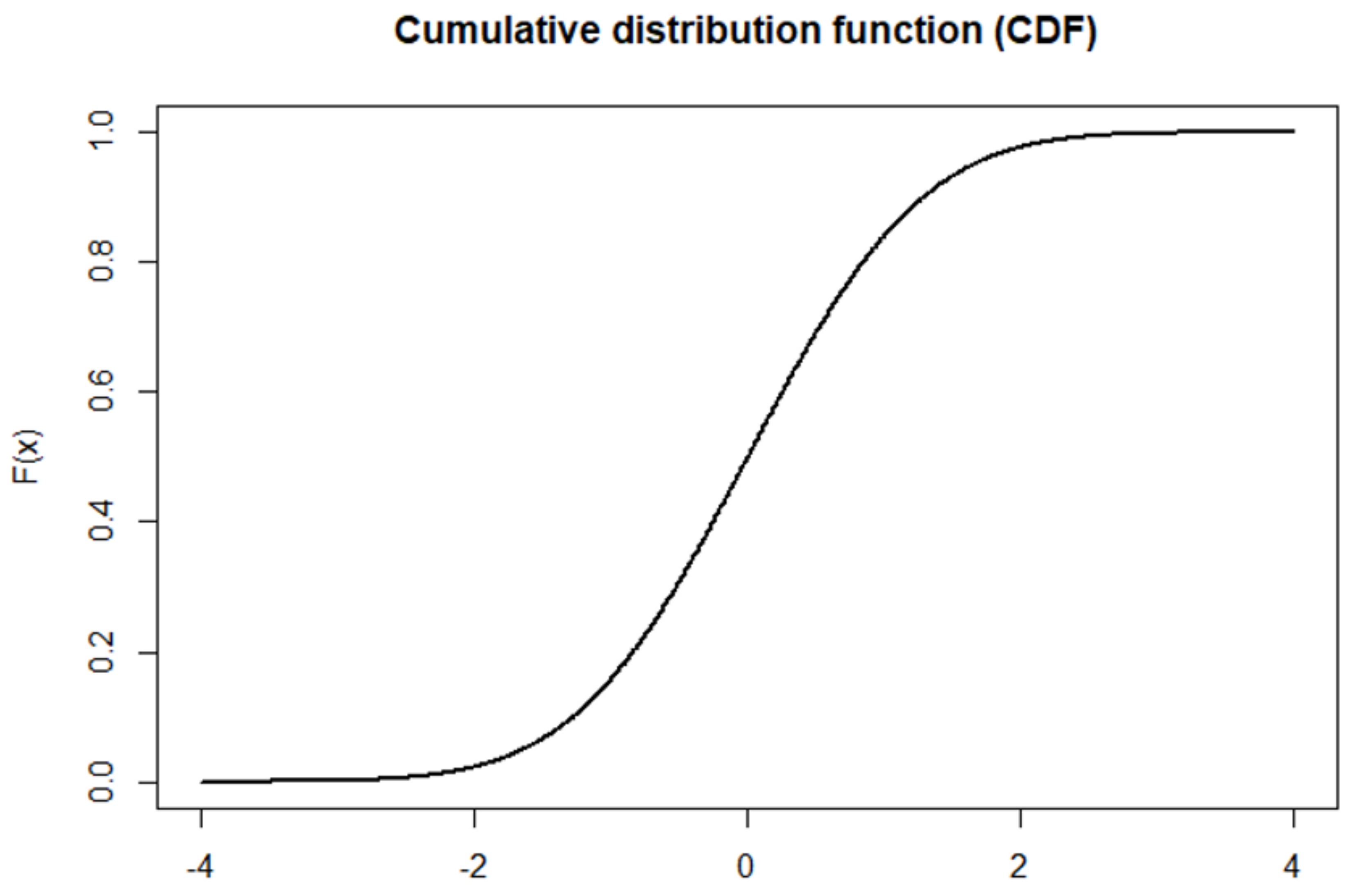


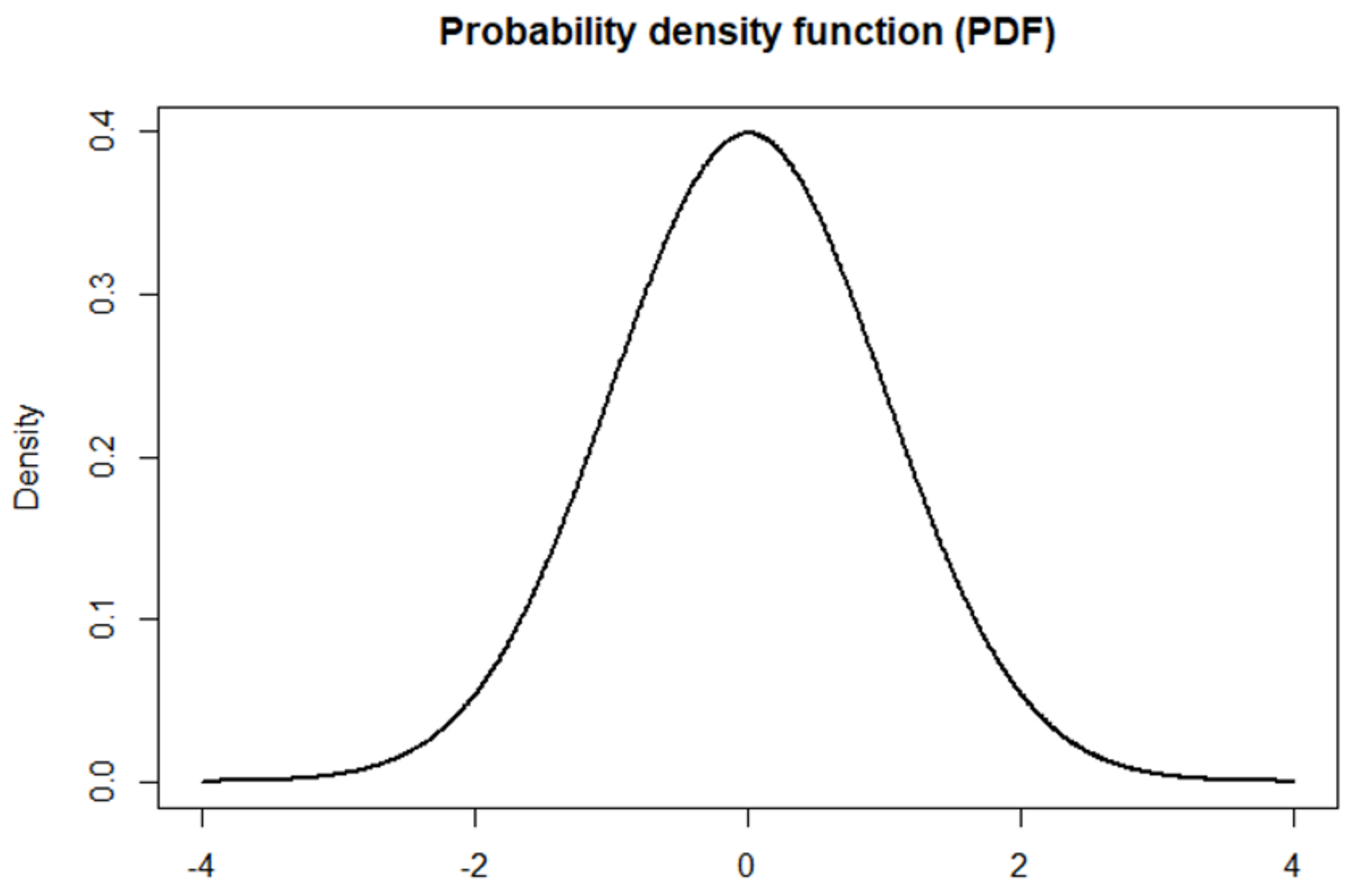


$$\mathcal{L}(f, x, y, \tau) = \begin{cases} \tau(y - f(x, \tau)) & y \geq f(x, \tau) \\ (\tau - 1)(y - f(x, \tau)) & y < f(x, \tau) \end{cases}$$

IMPLICIT QUANTILE NETWORKS LOSS FUNCTION







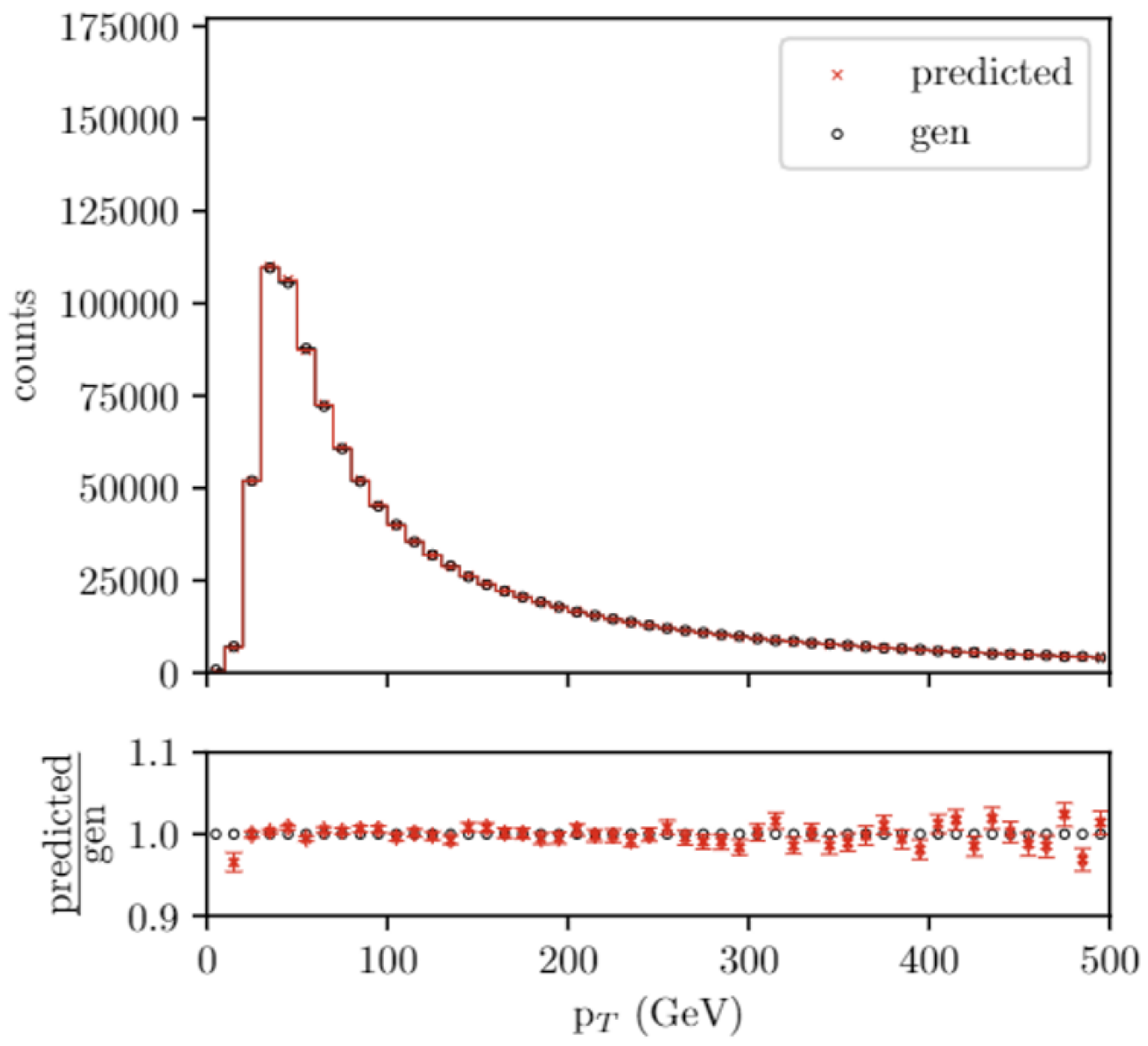
$$\mathcal{L}(f, x, y, \tau) = \begin{cases} \tau(y - f(x,\tau)) & y \geq f(x,\tau) \\ (\tau - 1)(y - f(x,\tau)) & y < f(x,\tau) \end{cases}$$

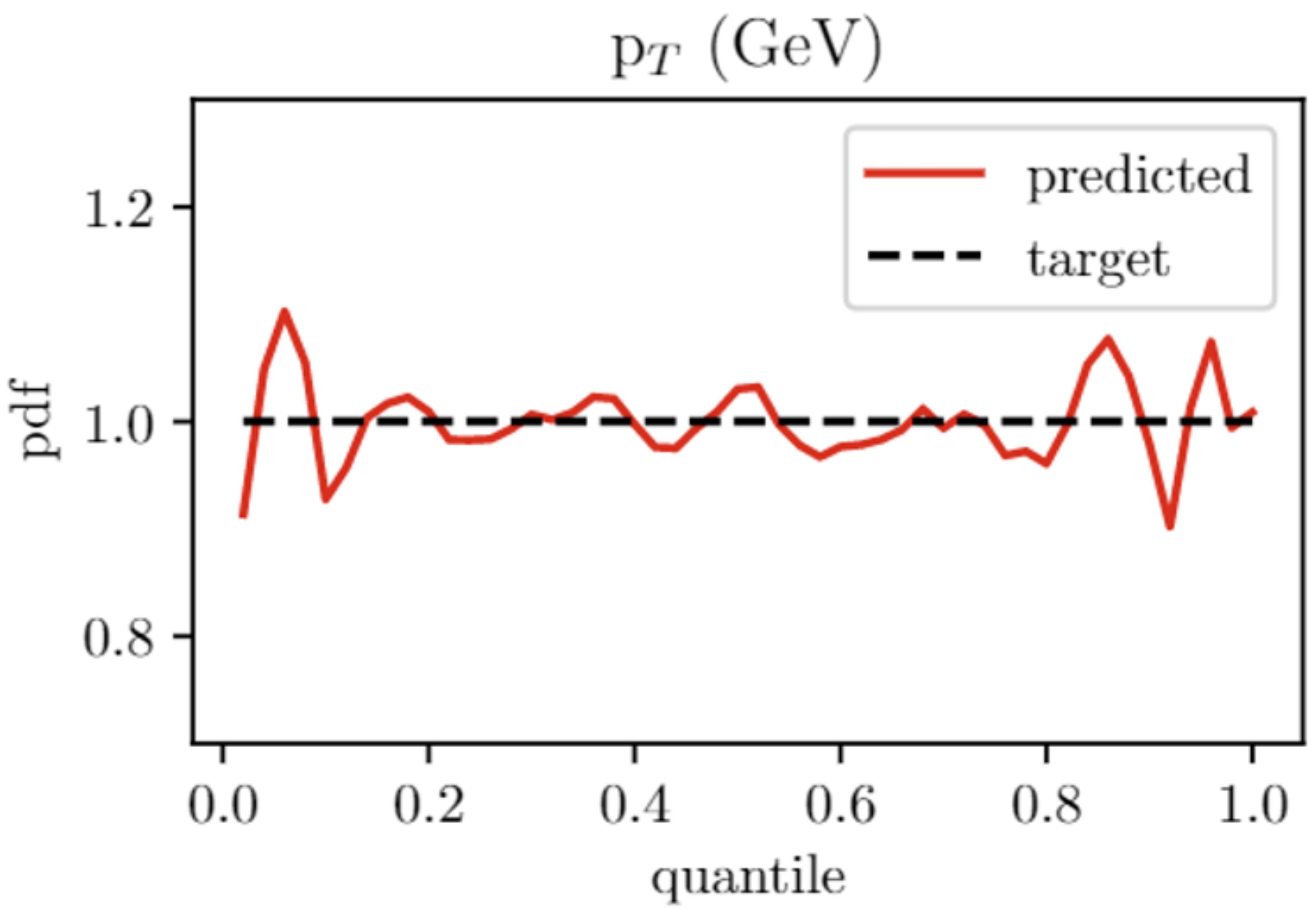
regularization

$$\begin{cases} \left(\frac{dy}{d\tau}\right)^2 & \frac{dy}{d\tau} < 0 \\ 0 & \frac{dy}{d\tau} \geq 0 \end{cases}$$

RESULTS JET CORRECTION

RESULTS JET CORRECTION





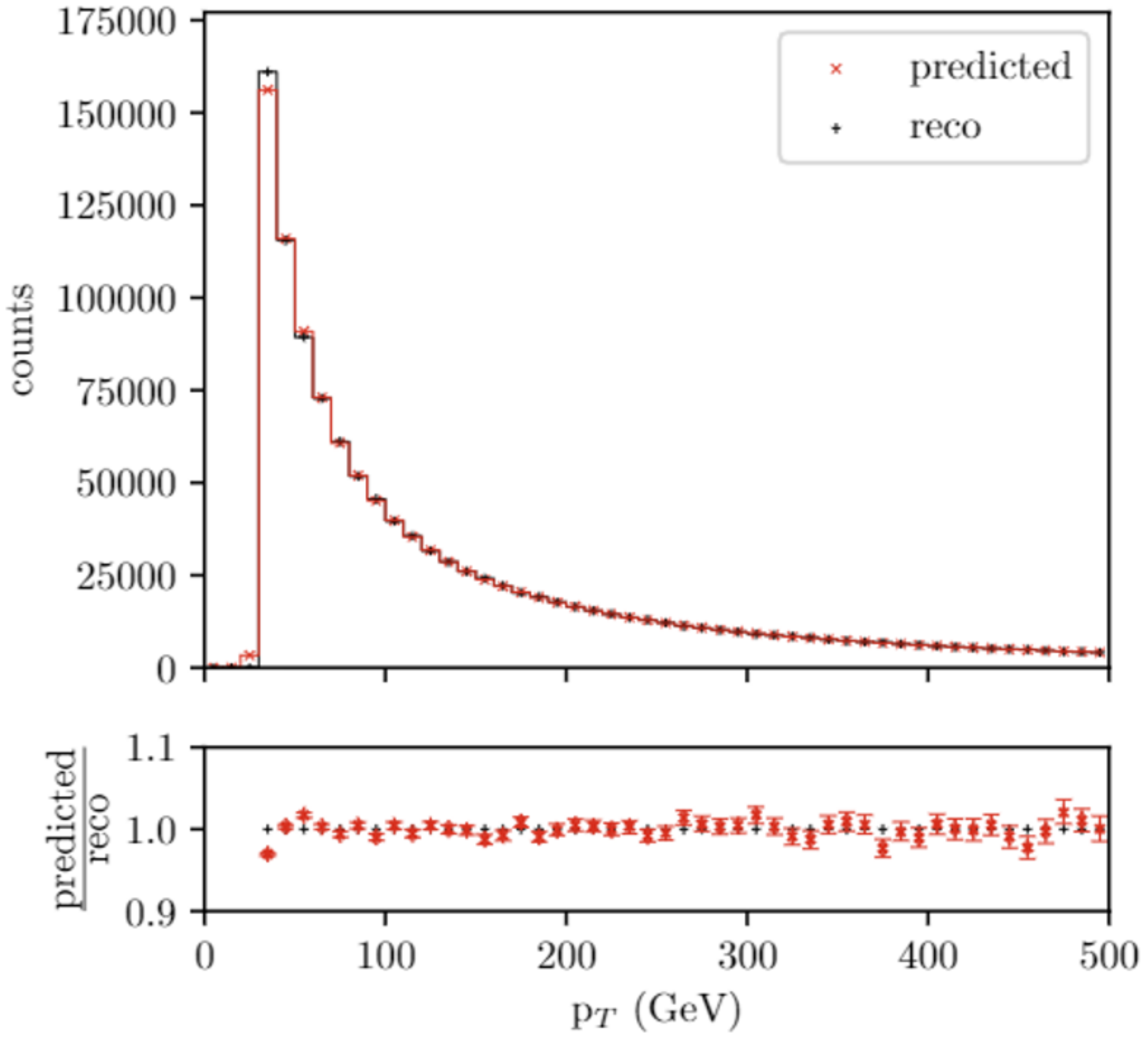
EV
GENER

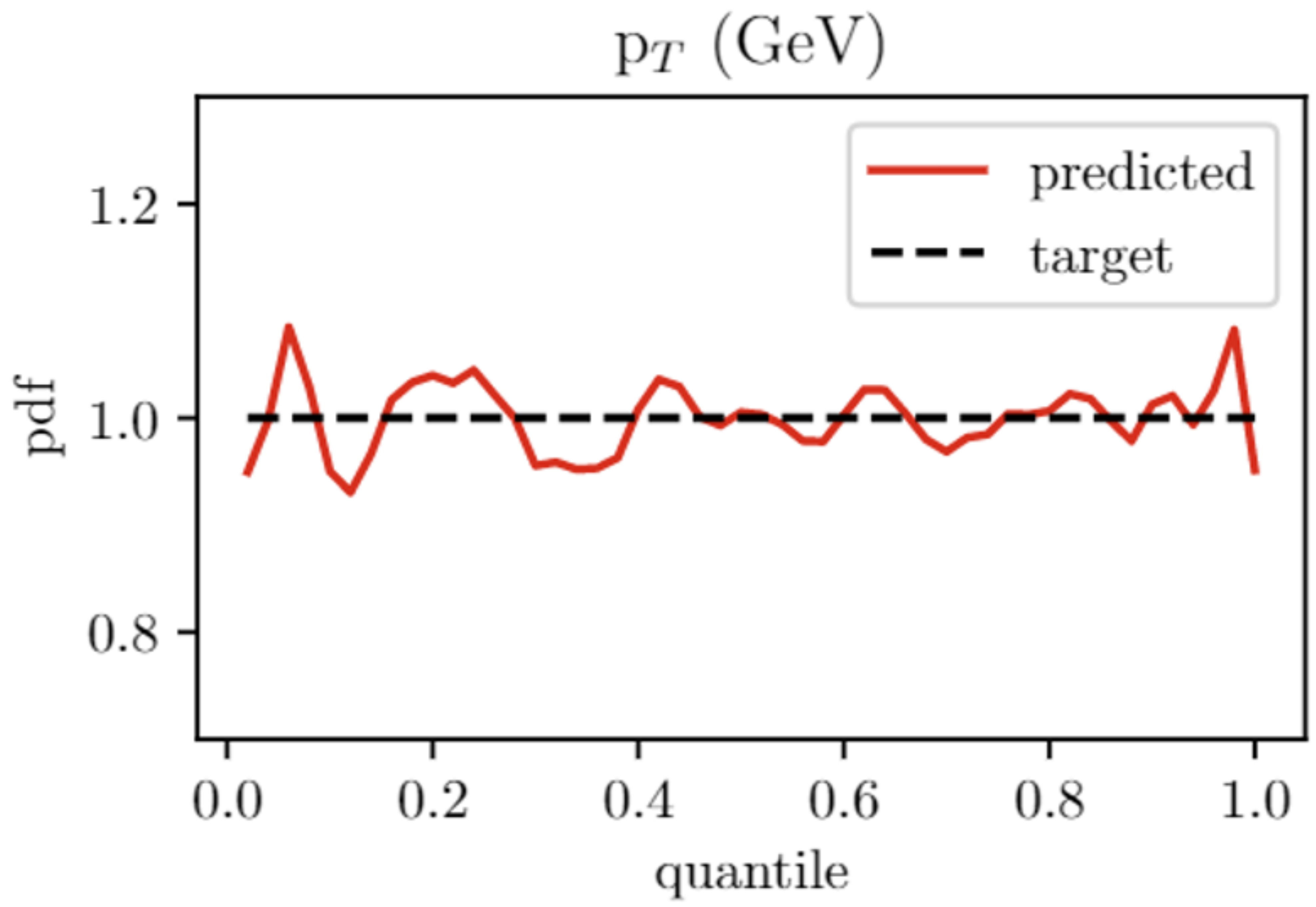
RECO
JETS

jet simulation

RESULTS JET SIMULATION

RESULTS JET SIMULATION





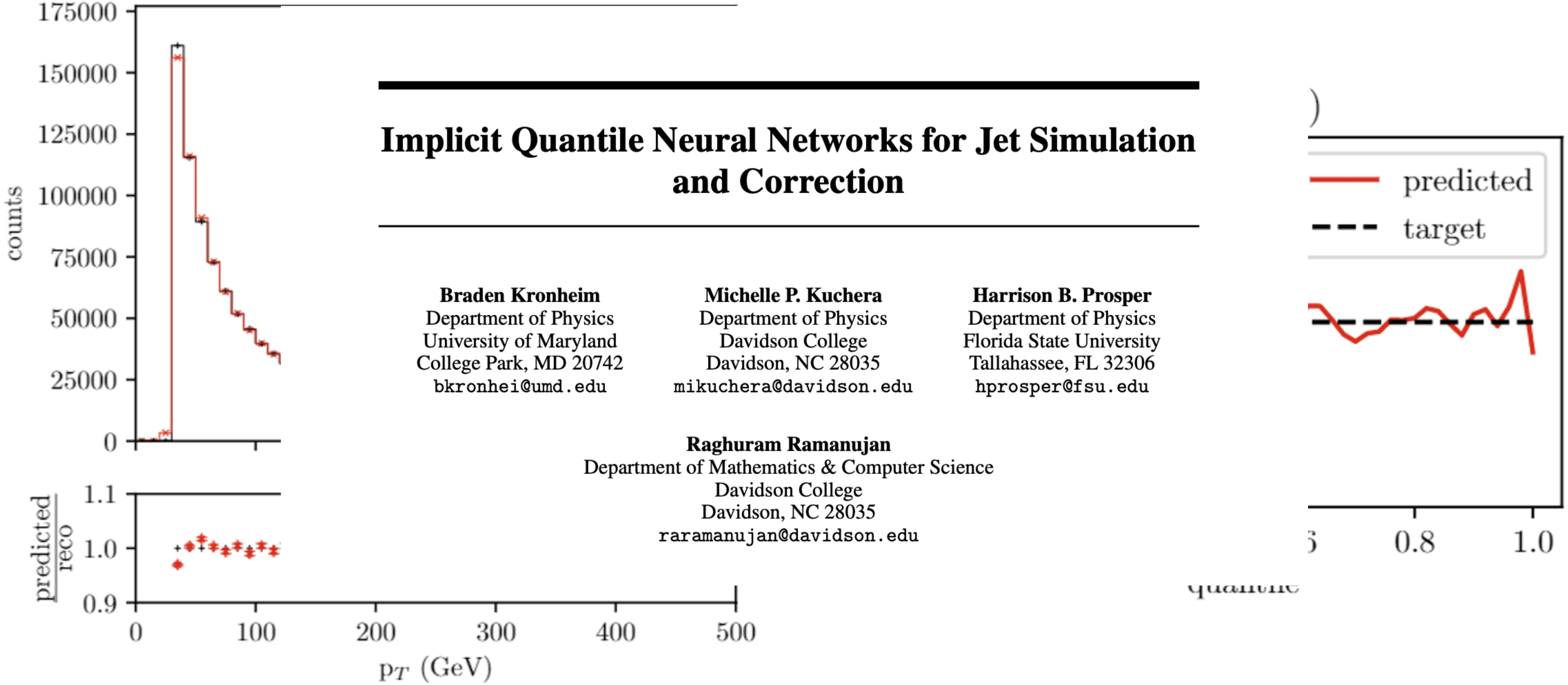
EVEN
GENERA

RECO
JETS

jet simulation

RESULTS JET SIMULATION

EVEN
GENERA



Implicit Quantile Neural Networks for Jet Simulation and Correction

Braden Kronheim
Department of Physics
University of Maryland
College Park, MD 20742
bkronhei@umd.edu

Michelle P. Kuchera
Department of Physics
Davidson College
Davidson, NC 28035
mikuchera@davidson.edu

Harrison B. Prosper
Department of Physics
Florida State University
Tallahassee, FL 32306
hprosper@fsu.edu

Raghuram Ramanujan
Department of Mathematics & Computer Science
Davidson College
Davidson, NC 28035
raramanujan@davidson.edu

RECO
JETS

jet simulation